NASA-CR-193220

Annual Report 1987

GRANT
IN-61-CR
171498
P. 70

RIACS

Research Institute for Advanced Computer Science

(NASA-CR-193220) THE 1987 RIACS ANNUAL REPORT (Research Inst. for Advanced Computer Science) 70 p

N94-11046

Unclas

G3/61 0171498

NASA Ames Research Center

1987
ANNUAL REPORT

Peter J. Denning
Director

June 20, 1987

Research Institute for Advanced Computer Science

The Research Institute for Advanced Computer Science
is supported by NASA under Cooperative Agreement No. NCC2-387.

TABLE OF CONTENTS

TABLE OF TABLES

1987
ANNUAL REPORT
Research Institute for Advanced Computer Science

Peter J. Denning
Director

June 20, 1988

The Research Institute for Advanced Computer Science (RIACS) was established at the NASA Ames Research Center in June of 1983. RIACS is privately operated by the Universities Space Research Association (USRA), a consortium of 64 universities with graduate programs in the aerospace sciences, under several Cooperative Agreements with NASA.

RIACS's goal is to provide preeminent leadership in basic and applied computer science research as partners in support of NASA's goals and missions. In pursuit of this goal, RIACS contributes to several of the grand challenges in science and engineering facing NASA:

- Flying an airplane inside a computer.
- Determining the chemical properties of materials under hostile conditions in the atmospheres of earth and the planets.
- Sending intelligent machines on unmanned space missions.
- Creating a one-world network that makes all scientific resources, including those in space, accessible to all the world's scientists.
- Providing intelligent computational support to all stages of the process of scientific investigation from problem formulation to results dissemination.
- Developing accurate global models for climatic behavior throughout the world.

In working with these challenges, we seek novel architectures, and novel ways to use them, that exploit the potential of parallel and distributed computation and make possible new functions that are beyond the current reach of computing machines. The investigation includes pattern computers as well as the more familiar numeric and symbolic computers, and it includes networked systems of resources distributed around the world. We believe that successful computer science research is interdisciplinary: it is driven by (and drives) important problems in other disciplines. We believe that research should be guided by a clear long-term vision with planned milestones. And we believe that our environment must foster and exploit innovation.

This document reports on our activities and accomplishments for the calendar year 1987 and our plans for 1988.

We at RIACS gratefully acknowledge the high level of technical and moral support we have received from Ames personnel.

1. RESEARCH REPORT

1.1. Overview

The projects of the RIACS research program have been grouped to reflect five major areas of collaboration between RIACS and NASA:

Advanced Algorithms and Architectures
Sparse Distributed Memory Studies
Telescience
Scientific Computing Environments
Applied Computer Science

In each area, RIACS personnel worked closely with NASA Ames personnel to achieve the stated goals.

The Advanced Algorithms and Architectures area seeks to understand the matches between novel computing architectures and scientific problems of interest to NASA, primarily computational fluid dynamics (CFD) and computational chemistry (CC). These investigations include the design and testing of algorithms capable of delivering the full power of the hardware to the problem, over a wide range of problem sizes -- i.e., of algorithms that scale in speed almost linearly as the number of processors in the machine increases. These studies will assess whether a particular architecture has potential as a supercomputer and, if so, will develop efficient new algorithms that solve kernel problems in CFD and CC on that architecture. Major support for these projects is provided by the Numerical Aerodynamic Simulation (NAS) Systems Division and by DARPA support of the Center for Advanced Architectures under an Interagency Agreement with NASA.

The Sparse Distributed Memory (SDM) stores patterns representing encoded sensory data and rapidly retrieves one of them when presented with a similar pattern. Patterns are stored as long binary vectors (e.g., 1000 bits) and a given address pattern will retrieve a stored pattern if it is close in Hamming distance. This project is advancing the theory of the SDM and is experimenting with applications in vision, speech, and robotics. Simulators of the memory have been programmed on the Intel hypercube, the Sequent Balance, and the Connection Machine 2 computers. A hardware prototype is being constructed through a collaboration with the CASIS project at Stanford University. Major support for this project is provided by the Aerospace Human Factors Division and the Computer Systems and Research Division.

Telescience is NASA's term for the conduct of scientific research with remote resources separated by wide distances -- e.g., servers, databases, instruments, and people. This project is concerned with the design of the networks and distributed computations required to make Telescience a reality in the Space Station era. It includes the remote design and installation of experiments on space platforms, the operation of those experiments, and the collaboration of scientists around the world in analyzing and publishing the results. The work of the project has included the design of a Telescience Workstation, a testbed for the Life Sciences, and an interdisciplinary testbed operated within a consortium of fifteen universities. Through the testbed, USRA and RIACS, together with a consortium of universities, have conducted experiments in several scientific disciplines to validate requirements and assess technologies for future telescience systems. This project has received major support from the Human Factors Division, the Advanced Technology and Space Station Planning Office, and a contract to USRA for the Telescience Testbed Pilot Program.

The Scientific Computing Environment is a workstation-based testbed of functions that can automate routine aspects of scientific investigation and seamlessly integrate processing, memory, networking, and software elements of high performance computing. It is an integral part of making the next generations of high performance computing systems usable and accessible to scientists. The types of functions included are: scientific office environment, interactive 3D and 4D graphics, symbolic mathematics, access to supercomputing resources, and distributed program composition. This project involves collaboration between RIACS and the Computational Chemistry Branch of Ames to develop video tape displays of potential energy fields that determine how molecules interact during a collision.

In addition to projects in the four main areas of collaboration, RIACS undertook several special projects in which a principal investigator carried out a program of research with sponsorship from a branch at the NASA Ames Research Center. These projects include network analysis, automatic inference of classes of data, high performance network architectures, technical advice for the NAS Systems Division, and computational chemistry. These projects have been grouped under the heading of Applied Computer Science.

Figure 1 is an organization chart showing the division of RIACS into project areas and the names of the persons associated with each area. RIACS is committed to making a difference in each of the project areas; the difference can take several forms, for example a shift in the way NASA or the government approaches an area, a commitment by NASA to use a system or result in a a future mission, or wide acknowledgement within the scientific community that a significant new line of investigation has been opened up. For this reason, RIACS maintains a balance in each area, with at least 25% of the effort in either basic research or applied research. It should be understood that references to RIACS here mean "RIACS acting in partnership with NASA."

Table 1 lists the projects under way in 1987 in pursuit of the visions outlined above. These projects are described in the sections following. Table 2 lists all personnel who were employed by RIACS in 1987. Table 3 summarizes how technical personnel and funding were divided among the project areas in 1987. The line marked "Total 1987 Grants" refers to the total of the budgets for these projects in 1987. (As reported later in Table 10, the 1987 expenditures for these projects were, in aggregate, slightly less than budget.) Publications resulting from the projects are listed in Appendix A.

RIACS Organization

Director
DENNING

Cramer

Deputy
LEVIN

Curlette
Kohutanycz
Lopez (c)

Chief Scientist
RAUGH

Senior Staff Scientist
LEINER

SDM
RAUGH

Danforth
Jaeckel
Joglekar
Kanerva
Keeler (s)
Olshausen
Rogers

Stewart

AAA
RAUGH

Adams (c)
Callahan (c)
Chan (c)
Tuminaro (s)
Golub (c)
Concus (c)

Stewart

CAA
LEVIN

TS
LEINER

Gallagher
R Johnson
V Johnson
Koch (c)
Shames (c)
Slocum
Young (c)

Fisher

SCE
BROWN

Gehrt
Klimbal
Gomez

ACS
LEVIN

Cheeseman
M Johnson
Levin
Ollikainen
Zweben

(c) consultant (s) student

1/88

TABLE 1
Principal RIACS Projects in 1987

NASA Number*	Project
ADVANCED ALGORITHMS AND ARCHITECTURES	
C	Parallel Multigrid Methods (*Chan, Tuminaro*)
C	Evaluating Supercomputers (*Denning, Adams*)
402	CRAY-2 Algorithm and Performance Studies (*Calahan*)
C	CFD on Teraflops Computer (*Raugh, Levin*)
SPARSE DISTRIBUTED MEMORY	
408	System Theory (*Kanerva*)
408	Large-Scale Simulations (*Rogers*)
408	Digital Prototype (*Flynn et al.*)
408	Information Capacity Studies (*Keeler*)
C, 408	Demonstrators (*Brown, Kanerva, Leong, Raugh*)
408	New Architectures (*Jaeckel*)
408	Data Tagging (*Rogers*)
408	Mathematical Studies (*Wang, Jaeckel*)
408	Optical Character Recognition (*Jaeckel*)
408	Text-to-Speech Translation (*Joglekar*)
408	Bibliography of Visual Encoding Techniques (*Olshausen*)
408	Speech Recognition Laboratory (*Danforth*)
408	Optoelectronic Cerebellum Network (*Loebner et al.*)
TELESCIENCE	
C	Telescience Workstation Environment (*Slocum*)
C, 427	Scientific Networking (*Leiner*)
C	Applied Telescience Testbedding (*V. Johnson*)
NASW-4234	Telescience Testbed Pilot Program (*Leiner et al.*)
SCIENTIFIC COMPUTING ENVIRONMENTS	
C	Distributed Program Composition (*Brown*)
C	Distributed Concurrent C Debugger (*Lynch, Brown*)
397	Computational Chemistry Graphics (*Gomez*)
C	Digital Video Generation (*Brown, Klimbal*)
C	Facility Upgrades (*Klimbal*)
APPLIED COMPUTER SCIENCE	
431	NAS Extended Operating Configuration (*Levin*)
431	Computational Chemistry (*Levin*)
C	Networking Technology (*Ollikainen*)
C, 480	Network Analysis (*M. Johnson*)
C	NASA CS Research Plan (*Raugh, M. Johnson*)
428	Planning and Learning Research (*Cheeseman*)
C	The Science of Computing (*Denning*)

*C denotes NCC2-387, the core; others denote NCC2-xxx

TABLE 2
RIACS Employees in 1987

NAME	EDUCATION	AREA	FROM	TO	POS'N
Peter Denning	PhD, EE, MIT, 1968	Computer Systems	6-83	--	Dir
Eugene Levin	PhD, Math, UCLA, 1955	Scientific Computing	6-83	--	Dep Dir
Barry Leiner	PhD, EE, Stanford, 1973	Distributed Systems	8-85	--	Sr St Sci
Michael Raugh	PhD, Math, Stanford, 1979	Math Models	1-85	--	Ch Sci
George Adams	PhD, EE, Purdue, 1984	Computer Architecture	8-83	--	Sci
Matt Bishop	PhD, CS, Purdue, 1984	Computer Systems	9-84	6-87	Sci
Bob Brown	PhD, CS, Purdue, 1988	Operating Systems	7-83	--	Sci
Peter Cheeseman	PhD, AI, Monash U, 1979	Artificial Intelligence	5-85	--	Sci
Douglas Danforth	PhD, Educ, Stanford, 1978	Speech Recognition	1-88	--	Sci
Julian Gomez	PhD, CS, Ohio State, 1985	Graphics	1-86	12-87	Sci
Marjory Johnson	PhD, Math, U Iowa, 1970	Networks	1-84	--	Sci
Pentti Kanerva	PhD, Phil, Stanford, 1984	Parallel Cognitive Sys	10-85	--	Sci
Ari Ollikainen	MS, Comp Ling, UCLA, pending	Networks	8-87	--	Sci
David Rogers	PhD, Chem, UCSC, 1984	Neural Networks	8-87	--	Sci
Don Calahan	PhD, EE, U Ill, 1960	Parallel Algorithms	10-84	--	C'tnt
Tony Chan	PhD, CS, Stanford, 1978	Parallel Systems	1-86	--	C'tnt
Erol Gelenbe	PhD, Math, U Paris, 1973	Computer Systems	7-87	9-87	Vis
Louis Jaeckel	PhD, Stat, UC Berkeley, 1969	Applied Mathematics	8-87	--	C'tnt
Richard Johnson	PhD, Physics, IU, 1956	General Science	11-87	--	Vis
Monte Zweben	MS, CS, Stanford, pending	Artificial Intelligence	8-87	--	Vis
Nancy Blachman	MS, Opr Res, UC Berkeley, 1979	Systems	8-85	3-87	RA
Umesh Joglekar	MS, EE, UCSB, pending	Neural Networks	11-86	8-87	ra
Vicki Johnson	MS, CE, Stanford, 1980	Applied Science	8-87	--	RA
Philip Klimbal	BS, CS, Purdue, 1985	Distributed Systems	4-87	--	RA
Harrison Leong	PhD, BIS, CalTech, 1986	Neural Systems	10-86	9-87	RA
Bruno Olshausen	MS, EE, Stanford, 1987	Neural Networks	11-87	--	RA
Michael Slocum	MS, CS, UC Davis, 1981	Distributed Systems	6-87	6-88	RA
Ronald Chrisley	BS, Symb Syst, Stanford, 1987	Technical Documentation	10-86	6-87	stud
Leo Dagum	MS, AeroAstro, Stanford, 1987	Particle Simulations	7-87	--	stud
James Keeler	PhD, Physics, UCSD, 1987	Neural Networks	7-86	--	stud
Will Kessinger	BS, IE, Stanford, 1988	Operations Research	7-87	--	stud
Fung Lee	PhD, EE, Stanford, pending	Architecture	5-87	8-87	stud
William Lynch	MS, EE, Stanford, pending	Distributed Systems	5-87	8-87	stud
Raymond Tuminaro	PhD, CS, Stanford, pending	numerical computation	2-86	--	stud
Avery Wang	MS, EE, Stanford, pending	Parallel Computation	5-87	8-87	stud
Sam Ciraulo	BA, Speech Comm, S J St, pending	Administration	10-84	8-87	Adm'tr
Barbara Curlette	BS, Org Behavior, U SF, 1986	Administration	10-87	--	Res Anlyst
Lorraine Fisher	BS, CS, SJ St, pending	Administration	1-88	--	Sec'y
Maria Gallagher	BA, Hist/Poli. Sci, S J St, 1971	Administration	4-86	--	Proj Coord'r
Kathryn Hawken-Cramer	BS, Chem, U AZ, pending	Administration	6-87	--	Exec Sec'y
Anne Kohutanycz	BS, B Adm, S J St, 1984	Administration	1-85	--	Sys Adm'tr
Armando Lopez	BA, Physics, UC Berkeley, 1951	Administration	8-87	--	C'tnt
Helen Stewart	BA, BusAdm, SJ St, pending	Administration	9-87	--	Sec'y

Dir	-- Director	Dep Dir	-- Deputy Director
Ch Sci	-- Chief Scientist	Sr St Sci	-- Senior Staff Scientist
Sci	-- Scientist	Vis	-- Visitor
RA	-- Research Associate	ra	-- Research Assistant
stud	-- Student	C'tnt	-- Consultant

TABLE 3
1987 Scientific Staff and Funding Sources for Projects

AREA	CORE AGREEMENT [NCC2-387]	SUPPLEMENTAL AGREEMENTS	
		Investigator	Project
Advanced Algorithms and Architectures	Adams Chan (c) Dagum (s) Denning (5%) Lee (s) Tuminaro (s)	Calahan Levin (20%)	NCC2-402 NCC2-431
Sparse Distributed Memory	Brown (5%) Dagum (s) Keeler (s) Wang (s)	Kanerva Jaeckel (c) Joglekar (ra) Leong (RA) Olshausen (RA) Rogers	NCC2-408 NCC2-408 NCC2-408 NCC2-408 NCC2-408 NCC2-408
Telescience	Bishop (20%) Leiner (20%) Johnson, V (20%)	Bishop (80%) Leiner (30%) Johnson, V (80%) Slocum Johnson, R.	NCC2-398 NASW-4234
Scientific Computing Environments	Brown (95%) Klimbal Gehrt Lynch (s)	Gomez	NCC2-397
Applied Computer Science	Johnson, M (20%) Ollikainen (20%) Denning (5%) Kessinger (s) Raugh (10%)	Johnson, M (80%) Ollikainen (80%) Cheeseman Levin (20%) Zweben	NCC2-480 NCC2-428 NCC2-431 NCC2-428
RIACS Management **Director** **Deputy Director** **Chief Scientist** **Senior Staff Scientist**	 Denning (90%) Levin (60%) Raugh (90%) Leiner (50%)		

1.2. Advanced Algorithms and Architectures

Project Leader: Michael Raugh

One of NASA's grand visions is the ability to fly an airplane inside a computer. This means that a complete numerical simulation of flows around an airfoil must be computable in near real time. Significant increases in computational power are required to achieve this; the best estimates are that the needed systems must be about 1000 times faster than today's fastest supercomputers. To reach this goal, we need gross hardware speed-up of at least this factor and algorithms technology capable of delivering most of the hardware power to the computational problem.

Today's large-scale scientific computing is performed on vector supercomputers such as the Cray-2, the Cray X-MP, the Cyber-205, or their Japanese counterparts. Because electromagnetic signals can travel at most one foot in a nanosecond, it is doubtful whether the extremely powerful pipeline or vector-processing units at the core of these machines can be improved in speed much beyond a factor of ten. Supercomputer manufacturers have accordingly increased the computing power of their machines by increasing the number of individual processing units on each system. The next generation of supercomputing architectures will include machines based on a large number of simple processors rather than a small number of powerful ones. Computing power several orders of magnitude greater than that of current supercomputers is achieveable only with massive parallelism.

Massively parallel machines present new challenges to algorithms designers. How can large problems be decomposed into components that can be worked on by separate processors? How is the data flow among the program components to be implemented in the machine? How can the load be kept balanced among all the processors, especially when the exact distribution of processing requirements among problem components is data dependent? How can the communication costs among processors be kept to a minimum? The answers to these questions are complicated by the existence of a variety of different massively parallel architectures, each with its own intercommunication structure. This raises the additional question of how to determine which architectures are best suited for given problem classes -- e.g., turbulence modeling or calculating self-consistent quantum fields.

The Advanced Algorithms and Architectures project is exploring these questions with multidisciplinary teams that bring computer scientists together with computational scientists in areas of interest to NASA.

IN 1987 a project plan was formulated and put into action, so that in 1988 the project will experience a sharp increase in its staffing and scope. Three full-time scientists will be supported through cooperative agreements from the NAS Systems Division to work on problems in parallel algorithms in the context of solutions to 3D Navier-Stokes flow problems. With additional support from DARPA, the project will undertake the Center for Advanced Architectures, which will perform studies of algorithms on a specific massively parallel architecture, the Connection Machine 2.

Parallel Multigrid Methods

Tony Chan
Ray Tuminaro
(*Core*)

The multigrid algorithm is a fast efficient method for the solution of many partial differential equations. It is based on using a series of neighborhoods of a grid point, a neighborhood being defined for given k as the set of grid points at distance 2^k along each dimension. By performing a few iterations in each neighborhood, information from the boundaries propagates optimally through the whole grid and convergence is achieved in minimal time.

The multigrid algorithm's data flow is nontrivial and is somewhat representative of other fast methods for solving partial differential equations. We have considered the problems associated with the parallelization of multigrid algorithms on hypercubes. The work has considered both model problems and realistic applications. From our previous work with model problems we concluded that a significant reduction in execution time can be obtained by an effective parallelization of a multigrid code on a hypercube. However, as the ratio of the number of processors to the problem (grid) size grows the efficient utilization of the machine diminishes. This problem does not appear significant on small processor systems (say less than 100 processors). The objectives for our work in 1987 was to extend our results in two principal areas.

1. Evaluate the performance of the parallel multigrid method for nontrivial problems of interest to scientists at NASA Ames.
2. Develop multigrid-like methods which will better utilize a hypercube when a large number of processors are available.

Part of this work (specifically point 1) was done in collaboration with Eric Barszcz and Dennis Jespersen at NASA Ames.

To investigate the first problem mentioned above, we developed a code on the intel iPSC which solves the two dimensional steady state Euler equations over an airfoil. This code was modeled after Anthony Jameson's well known code FLO52. Our code has been used to produce timing results on the intel iPSC and an analysis of the intel iPSC's performance in solving realistic fluid dynamics problems. A poster session presenting our preliminary results was given at the SIAM Conference on Parallel Processing held in December 1987. In addition, we have considered modifications of the basic multigrid algorithm to effectively utilize a hypercube with a large number of processors. The usual multigrid algorithm suffers from a load imbalance on massively parallel machines because on coarser grids many processors may become idle. We have been investigating new algorithms for utilizing these idle processors to speed up the convergence of the iteration. Our main idea is based on filtering: we filter the usual residual and pass on only part of the frequencies to the coarser grid leaving another part to be iterated concurrently by the fine grid. We are also studying a new "superconvergent" algorithm recently proposed by Federickson and McBryan. A paper describing our algorithm and its effectiveness on certain model problems was presented at the Third Copper Mountain Conference on Multigrid Methods in April, 1987. We are currently trying this algorithm out on more difficult problems and comparing it with other approaches which are being developed.

We are currently extending our 1987 work in the following ways:

1. More analysis on our FLO52-like code. This includes running it on the new Intel iPSC/2, as well on an N-cube system. We hope to run on Sandia's N-

cube system with 1024 processors to investigate the performance of the code on a massively parallel machine. This code is attractive because it gives a practical testbed in which to evaluate new results in parallel multigrid algorithms.

2. Analysis and comparison of our multigrid-like method to effectively utilize large numbers of processors. This includes running our method on more problems and performing detailed comparisons with other methods.

We have also worked on parallel FFT algorithms. The least-communication for FFT is not the same as for nearest-neighbor grids. Thus in PDE computations one must be careful about the tradeoff between remapping and a suboptimal mapping. Chan gave an invited talk on this subject at a workshop on Parallel FFT Methods at Cornell University in March 1987.

Our final area of concentration was parallel domain decomposition methods for elliptic PDEs. These are coarse grain parallel methods suitable for coarse grain architectures such as multiprocessor Crays. The physical domain is decomposed into several smaller subdomains and the solutions on these are pieced together at the interfaces to form the overall solution, usually in an iterative fashion. Chan's work in this area has been primarily in the development of efficient preconditioners to speed up the convergence rate of this iteration. Several UCLA reports are available on this topic and Chan will continue work in this area with David Keyes of Yale Univ at RIACS in summer 1988.

Publications and Presentations:

1. Eric Barszcz, Tony Chan, Dennis Jespersen, and Ray Tuminaro, "Solution of the Euler Equations on a Hypercube," poster presentation at the SIAM Meeting on Parallel Processing in Los Angeles, Dec., 1987.

2. Tony Chan and Ray Tuminaro, "Design and Implementation of Parallel Multigrid Algorithms," appears as RIACS Technical Report 87.21, Aug., 1987, and also in The Proceedings of the Third Copper Mountain Conference on Multigrid Methods, S. McCormick (ed), Marcel Kekker, NY, 1987.

3. Tony Chan and Ray Tuminaro, "A Survey of Parallel Multigrid Methods," appears as RIACS Technical Report 87.22 and in the Proceedings of the American Society of Mechanical Engineers Winter Annual Meeting,1987 in a book titled Symposium on Parallel Computations and Their Impact on Mechanics, Dec., 1987.

4. Eric Barszcz, Tony Chan, Dennis Jespersen, and Ray Tuminaro, "Evaluation of the Parallel Solution of the Euler Equations on a Hypercube Architecture," in preparation.

Research Questions for Evaluating Supercomputers

George Adams
Peter Denning
(*Core*)

Supercomputing systems lie at the frontier of computing technology and scientific inquiry. A careful examination of the scientific accomplishments of the performance analysis field reveals that methods and models capable of dealing with parallel and distributed supercomputation are incomplete and inadequate. Supercomputing offers many challenges for performance analysts. Under the heading of speed, the challenges concern analytic and simulation models. Under the heading of usability, the challenges concern matches between problem domains and architectures, as well as new tools for supporting a domain's problem solving process. Far less progress has been made under the usability heading than under the speed heading. We identified research needs in the areas listed below. The results of this study were presented at the International Seminar on Large Scale Scientific Computing, INRIA, Rocquencourt, France, in February, 1987, and were published in *Supercomputing State of the Art* by North-Holland (1987).

- SPEED
  - Analytic models (predictive)
    - parallelism
    - workloads (models and benchmarks)
    - measurement tools
  - Simulation models
  - Modification models
  - Model validation
- USABILITY
  - Problem solving processes
  - Tools
    - current needs: support distributed computation
    - future needs: support stages of problem solving
    - quality control
  - Domain-architecture matches
    - formal descriptions
    - domain patterns (top-down)
    - architecture patterns (bottom-up)
    - heuristics for selecting architectures
    - automatic selection of appropriate architecture
  - Deep evaluation studies in domains

CRAY-2 Algorithm and Performance Studies

Don Calahan
(*NCC2-402*)

The CRAY-2 (C-2) represents the first in a series of research machines by Cray Research, Inc. (CRI), to explore new architectural concepts related to the technology associated with high-performance low-parallelism scientific processors. These machines introduce unique algorithm and performance problems being studied in this research.

ALGORITHM STUDIES. These have been reported in previous annual reports and will not be considered here.

MEMORY PERFORMANCE STUDIES. Conflicts associated with reads and writes from the slow main memory have been found to reduce performance of many library and production codes by up to 50% in early dynamic-memory C-2s. It is likely that most production codes achieve only 2- to 3-processor performance from such a 4-processor C-2.

Dedicated experimental tests of three C-2's -- two at ARC with dynamic memory and one at CRI with static memory -- with have been made to model those load characteristics which degrade a user code. At present, a mathematical model derived from empirical results is being developed which includes two load parameters: 1) the rate at which memory accesses are made (accesses/cp), irrespective of their origin, and 2) the rate at which new vector accesses are made (vector startups/cp). The latter models the effects of accessing pattern irregularities.

The goal of these studies is to develop a mathematical model and an associated set of software measurement probes by which one can 1) model the load of present C-2 sites, 2) predict and verify the degradation of codes at these sites (the code access sensitivity problem), and 3) predict degradation associated with a given memory technology and design. The latter, a long-term goal, would require simulation, and would be intended to influence design of future CRI machines.

Potential CFD Applications on Teraflops Computer

Michael Raugh
Gene Levin
(*Core*)

NASA seeks a teraflops computer by the mid 1990s to enable it to simulate full-body transonic flow. A group of researchers at the IBM Thomas J. Watson Research Center announced in 1987 that they are developing an ultra-high performance, massively parallel scientific computer, called the TF-1. This name signifies a peak performance in excess of one teraflops. NASA, IBM, and RIACS are discussing the possibility of cooperating in the development of the TF-1, with IBM producing the hardware and NASA/RIACS producing an operating system, language, and two important applications in computational fluid dynamics. Informal studies were initiated in 1987 and will be continued in 1988 to determine whether there is a basis for a formal cooperation.

1.3. Sparse Distributed Memory

Principal Investigator: Pentti Kanerva
Project Manager: Michael Raugh

One of NASA's grand challenges is machines and systems that are capable of learning to function in places and tasks too remote, too hostile, or too tedious for humans. The Sparse Distributed Memory (SDM) project is investigating the theory and applications of a massively parallel computing architecture that will support the storage and retrieval of sensor and motor patterns characteristic of autonomous systems. The immediate objectives of the project are centered in studies of the memory itself and in the use of the memory to solve problems in speech, vision, and robotics. Investigation of methods for encoding sensory data is an important part of the research. Examples of NASA missions that may benefit from this work are: Space Station, planetary probes, and solar exploration. Sparse distributed memory offers promising technology for operating any large complex system requiring automatic monitoring and control.

Sparse distributed memory is a massively parallel architecture motivated by efforts to understand how the human brain works, given that the brain comprises billions of sparsely interconnected neurons, and by the desire to build machines capable of similar behavior. Unlike numerical and symbolic computers, sparse distributed memory is a pattern computer, designed to process very large patterns formulated as bit strings that may be thousands of bits long. Each such bit string can serve as both content and address within the memory. Our model of an autonomous system has four main components: a "focus" that holds a large pattern of features representing a moment of the system's experience, a sensory apparatus that extracts features from signals received from the world and feeds them into the focus, a motor apparatus that is driven from the focus and affects the world, and a long-term memory for large patterns that is connected to the focus and holds an internal model of the world built at least in part through experience.

Sparse distributed memory is a member of the family of connectionist architectures. It differs from the related family of neural-net architectures because it is capable of storing very long pattern sequences as well as individual patterns. Therefore, the memory can store a dynamic model of the world that can be used for prediction based on experience. The sparse distributed memory can simulate the ordinary computer memory (RAM) but, being optimized for long patterns stored in multiple locations, cannot give the same speed for storage and retreival.

Sparse distributed memory was invented by the principal investigator, Pentti Kanerva [*Sparse Distributed Memory*, MIT Press (in press)]. A digital prototype of sparse distributed memory is being developed at Stanford in cooperation with this project and with the support of NASA. The Sparse Distributed Memory project is being carried out in cooperation with NASA scientists working in vision and human factors in space flight, and it will make it possible to decide whether and how to build sparse distributed memories for real tasks.

System Theory

Pentti Kanerva
(*NCC2-408*)

Sparse distributed memory was developed as a model of human long-term memory. In our modeling, a large pattern of features encodes a moment of a system's experience, and a sequence of such patterns models the system's experience over time. Because the patterns stored in memory can be used also to address the memory, sequences can be stored as pointer chains. Any pattern in a stored sequence--or a pattern that is sufficiently similar to some pattern in a stored sequence--can then be used to retrieve the rest of the sequence. The stored sequences can be arbitrarily long because the capacity of the memory can be made arbitrarily large.

In human cognition, memory plays but a part, however important: It stores a dynamic, predictive model of the world. Another part has to do with extracting information from the world and encoding it in forms suitable for storage in memory. That part is carried out by the senses of sight, hearing, touch, and others with assistance from memory. The function of a sensory system may be summarized as follows: The memory works with features and creates internal objects and individuals by chunking together things that are similar in terms of those features. For the internal objects to match objects of the world, the system's sensors need to transform raw input from the world into features that are relatively invariant over natural perturbations of objects. To recall a stored object, the senses--or the memory--have to produce a reasonable approximation of the encoding that was used as an address when the object was stored.

Yet another part of human cognition has to do with action as a way of affecting the world. Actions are carried out by a system's motors or muscles. In our modeling of motor systems, the motors are controlled by sequences of patterns that can be stored in memory.

These ideas have been combined in a simple model of an autonomous learning system. The system has a central place called the focus that accounts for the system's subjective experience. The entity in the focus is a very large pattern that encodes everything about that moment--that is, any specific things that the system is attending to, the system's action, and the overall context. The memory is addressed by the focus, the memory's output goes into the focus, the senses feed into the focus, and the motors are driven from the focus. This architecture has been motivated by the oneness of subjective experience: An experience created by the senses can be created also from memory. The system's modeling of the world is founded on this idea.

A system with such an architecture should be capable of learning how the world works and how its own actions affect the world, including how they affect its own well-being. The well-being is modeled by a built-in preference function that is defined on the states of the focus. In learning to act, the system needs to store favorable action sequences in memory and to assign positive and negative preferences to previously indifferent states. In the most advanced form of learning--namely, imitation--the system uses itself to model the behavior of others of its kind.

Report of this work will appear in 1988 and will be included in Kanerva's forthcoming book *Sparse Distributed Memory* (MIT Press).

Large-Scale Simulations

David Rogers
(*Core*)

Kanerva's original model of sparse distributed memory used 1000-bit addresses and allowed for the storage of 1000-bit data words (RIACS TR 86.14, A View of Kanerva's Sparse Distributed Memory). In such a model, any 1000-bit word may be used as either *content* (data) to be stored in the memory or as *address* in the memory. A natural way to store a *sequence* of data is to encode each datum of the sequence as a 1000-bit number and regard the number as an address. The sequence may then be stored as a pointer chain -- that is, each storage location contains the address of (hence the code for) the next datum of the sequence. An obvious problem is that the scheme breaks down in the case of two distinct sequences that share a common data point. To avoid this problem, sparse distributed memory incorporates *folds*. A k-fold memory stores the context of a sequence, in the sense that at any point in the sequence it takes account of the k immediately preceeding items. In this way it is possible to "look back" in the sequence for as many as k steps; thus two overlapping sequences will be distinguished so long as the overlap is fewer than k successive items.

A simulator of SDM was designed for a Thinking Machines CM-2, a massively parallel SIMD computer with 16K processing elements and 128 megabytes of memory. This work will allow simulating a k-fold SDM having as many as 250,000 physical memory locations with 256-bit addresses and having a performance of approximately five memory cycles per second. (The same software, running on a full-size CM-2 with 64K processors, can simulate an SDM with 1 million physical memory locations.) The simulator will be used to test design parameters and to study applications of SDM to problems in speech and vision. One example of a problem we are considering is that of detecting word boundaries in continuous speech, in which an extended utterance is encoded as a sequence of phoneme-like units derived from lower-order processing. Subsequences that occur with high frequency will provide approximations of words; folds will be used to determine word boundaries as well as to distinguish between words with common phoneme groups. A second example will test the effect of varying the local density of the memory. In the original model physical memory locations are spread uniformly throughout the address space, and the Hamming radius for reading and writing is held constant. But in order to deal with highly correlated data, it may be more efficient to vary the density as a function of locality in accord with the density of the data and to likewise vary the radius inversely with the data density. We plan to investigate this approach to storing data for discrete speech recognition.

In 1988 software and a report describing the simulator and how to use it will be provided. Experiments on error-recovery properties as a function of pattern packing in the memory will be performed and reported.

Digital Prototype of SDM

Michael J. Flynn (Stanford)
Pentti Kanerva (RIACS)
B. Ahanin (Stanford)
N. Bhadkamkar (Stanford)
P. Flaherty (Stanford)
P. Hickey (Stanford)
(*NCC2-408*)

The mathematical theory of sparse distributed memory is complete and has been verified by computer simulation. The theory is also practical in that memories based on it can be implemented with conventional RAM-memory elements.

Stanford's Computer Systems Laboratory, working in consultation with the RIACS SDM group, completed design specifications for an SDM digital prototype. A sparse distributed memory, faithful to the model described in Kanerva's dissertation (CSLI report 84-7), has been built and is undergoing testing at Stanford. The memory array in the prototype memory makes extensive use of 1 M-bit DRAM technology, with array modules in concurrent execution. Consequently, the prototype is inexpensive compared to implementations of the memory on systolic-array, "connection machine," or general-purpose equipment. A copy of the Stanford prototype SDM is incorporated into a Sun workstation as an attached processor. The prototype, designed to perform fifty memory retrievals per second, has an address length of 256 bits and initially has 8,000 physical memory cells but can be upgraded to 80,000 memory cells. The prototype will be used at RIACS for a variety of experiments, including recognition of discrete speech, translation of English text to phonemes, and control of a hand-eye coordination system. A report describing the prototype has been published by Stanford University (Technical Report CSL-TR-87-338, Sparse, Distributed Memory Prototype: Principles of Operation; also available as RIACS TR 88.12).

Information Capacity Studies

James Keeler
(*NCC2-408*)

The digital prototype of sparse distributed memory being constructed at Stanford uses 256-bit addresses; memories with much larger addresses, say thousands of bits long, are feasible and will be practical to build in the near future. The address spaces of such memories are astronomical in size, hence any implementation will necessarily use only a very sparse subset of possible addresses for hardware locations. One implication of this is that, given such an address size, sparse distributed memories may be built large enough to store any realistic amount of data. The question arises whether, for a given address size, the number of hardware locations has a degrading effect on storage efficiency (i.e., capacity per hard location) as the number of hardware locations is increased. We have learned that there is no such degradation: The information capacity of sparse distributed memory is proportional to the number of hard locations (Keeler, J.D., Information Capacity of Outer-Product Neural Networks. *Physics Letters*, A 124, 53-58.)

A problem with associative memories like the Hopfield model and the SDM is that they can only store randomly-distributed patterns -- they are inefficient for storing correlated data. A minor modification of the SDM is to choose the addresses from the input distribution and dynamically adjust the Hamming radius to keep the number of selected locations constant. This modification allows storage of non-random, correlated patterns as was shown by Keeler (Keeler, J.D, "Capacity for patterns and sequences in Kanerva's SDM as compared to other associative memories." Proceedings of the conference on Neural Information Processing Systems, Denver (1987)). An analog address decoder and feedback circuit was designed for dynamically adjusting the Hamming radius.

Demonstrators

Robert Brown
Pentti Kanerva
Harrison Leong
Michael Raugh
(*Core, NCC2-408*)

In early 1987, an instructional videotape, "Sparse, Distributed Memory: How Does It Work?" was completed to demonstrate the memory's dynamic behavior. A narrative description of the memory is coordinated with pictures of the memory array, showing an address bank, a select vector, the data bank, input registers for address and data patterns, and an output register for a data pattern. Visualization is enhanced by color displays generated on a Sun workstation by an SDM demonstrator described in RIACS TR87.17, *Two Demonstrators and Simulator for a Sparse Distributed Memory*.

New Pattern-Computer Architectures

Louis Jaeckel
(*NCC2-408*)

A variety of extensions of the SDM architecture have been designed for improving performance, for dealing with correlated data, and for use in specific application areas, such as optical character recognition. This work will be reported in detail in 1988 after a patent application has been filed.

Data Tagging

David Rogers
(*NCC2-408*)

The original theory of sparse distributed memory was based on the assumptions that hardware memory cells are spread uniformly throughout the address space and that data are likewise stored at evenly distributed random locations (see RIACS TR 86.2, Parallel Structures in Human and Computer Memory). In real applications, however, data are highly correlated as are the addresses at which the data are stored. As a result, performance of the memory on real data sets is not as efficient as on uncorrelated data sets. The standard formulation of sparse distributed memory involves the selection of a "sphere" of data storage locations within a given Hamming distance of the specified address, followed by averaging the data contained in those locations to reconstruct the stored data.

This work involves a modification of the standard formulation of SDM, incorporating additional bits concatenated to each storage location for recording *data tags* with each datum. The use of tags permits estimating signal-to-noise factors for each storage location and using the factors for weighted averaging of data for retrieval. Experimental results show that significant improvements in recall capability can be achieved, especially for correlated data. For one example, it was shown that mildly correlated data could be recovered with an 80% improvement in the number of bits correct when compared to performance of a standard sparse distributed memory. (See RIACS TR-88.1, Using data tagging to improve the performance of Kanerva's sparse distributed memory.)

Further studies are being planned to determine the effect of tagging on specific correlated data sets, such as discrete speech, and on the ability of the memory to generalize. We also want to compare the effects of tagging with the effects of classical coding methods for improving information retrieval.

Mathematical Studies

Avery Wang
Louis Jaeckel
(*NCC2-408*)

In the original model of sparse distributed memory, reading and writing at an address involve all of the neighboring physical memory locations within a fixed Hamming radius of the address (See RIACS TR86.14, A View of Kanerva's Sparse Distributed Memory). In writing to an address, the datum to be stored is added to all of the neighboring locations. In reading from an address, all of the data from the neighboring addresses are pooled and thresholded. Because a datum is stored in many locations, it is possible to read it back again even when using an inaccurate address. It is only necessary that the pooled data surrounding the read address contain enough samples of the datum to drown out noise. Thus, the statistical properties of storage and retrieval depend on combinatoric results concerning the number of memory locations within a fixed Hamming distance from two separate memory addresses. Kanerva gave formulas for such numbers, which he derived by calculating the volume of intersection of two spheres of equal radius in very-high-dimensional vector spaces.

But it is possible to vary the Hamming radius used in addressing, and in particular it is possible to use a different radius for reading than for writing. This possibility is important because variation of the radius may improve information retrieval, especially in dealing with correlated data. In order to determine the statistical properties of storage and retrieval in this case, it is necessary to generalize Kanerva's formulas. This has been done by calculating the volume of intersection of two spheres of unequal radii. The new results help to provide a theoretical understanding of the performance properties of sparse distributed memory when the Hamming radius for writing data to the memory differs from the Hamming radius used for reading data from the memory. These results are also important because we believe that reading with a larger Hamming radius than the write radius is one way of abstracting general properties from correlated data. (See RIACS TR-88.5, Two Alternate Proofs of Wang's Lune Formula for Sparse Distributed Memory and an Integral Approximation.)

Optical Character Recognition

Louis Jaeckel
(*NCC2-408*)

Sparse distributed memory provides a means whereby a robot or autonomous system may develop a model of the world in which it operates and may learn from experience how to take actions suitable for carrying out the functions for which it was designed. Such a system requires sensors to sample the state of the world and motors to permit the system to act. To build practical systems based on sparse distributed memory it is necessary to learn how to encode sensory data in ways that take advantage of the memory's capabilities. The encoding of visual imagery is one example.

We have chosen to begin with the problem of optical character recognition because characters are composed of a well-defined set of simple components and the set of possible characters is very large. Our objective is to gain insight into principles for encoding more complex visual information, then to use those principles to solve more difficult problems of object recognition. As in speech recognition, the method of encoding the data appropriately for use in a sparse distributed memory is crucial and will be an important focus of effort.

Several simplifying assumptions are made at the outset. For one, it is supposed that a preprocessor has scanned the target text and analyzed each character into components that are either line segments or circular arcs. For another, it is supposed that the output of the preprocessor is continuous, in the sense that a slightly perturbed character is represented by components that are very similar to the components of the unperturbed character. Thus, we begin by assuming that each character of the text is presented as a set of well-determined components. At this stage of effort we do not concern ourselves with the problems inherent in designing the preprocessor, a plausible subject of research in its own right. Thus, the problem that we pose is to use sparse distributed memory to recognize characters when they are each presented as an unordered set of components.

Two distinct methods have been developed for encoding characters. In each of them the concept of *continuity of encoding* is very important. Characters (and hence components) must be encoded in such a way that slight variations in the character result in only slight changes in the code for the character. Since the character codes are used to make up addresses in a sparse distributed memory, we want the code of a perturbed character to lead to an address in the memory close to the address of the unperturbed character. This property is referred to as continuity of encoding.

In both of the encoding methods the technique for representing characters is the same. First, each character is *normalized* by translating it to the origin and scaling it. Then the character is decomposed into line segments and arcs of circles, each of which is represented uniquely by a set of five parameters. The parametrization is such that the parameters change continuously as the components are changed continously, and the parameterization for circular arcs is consistent with the parametrization for line segments in the sense that nearby arcs and segments are represented by nearly the same parameters. A segment or arc parametrized in this way is called a *generalized arc*, or simply *arc*. Thus an arc may be represented by a point on a bounded five-dimensional surface, which, surprisingly, has a twist like a Moebius strip. Finally, because the character is just an unordered set of arcs, it may be represented as the corresponding (unordered) set of points on the Moebius strip. The two encoding methods use different strategies for mapping these sets of points on the Moebius strip onto points in the address space of a sparse distributed memory.

Demonstration systems for both encoding methods are being designed and implemented. A prototype simulator has been written. A report describing both encoding methods is in preparation, and work investigating preprocessing methods for decomposing characters into appropriate parts, such as arcs, is being planned.

Translation From Text to Phonemic English

Umesh Joglekar
(*NCC2-408*)

Reading aloud from printed text is among the problems that cannot be easily solved by conventional computing methods. The widely publicized NETtalk program for translating written English into phonemes (Sejnowski and Rosenberg, 1986) demonstrated that it is possible for a parallel network of computing units to be trained to form internal representations of the regularities in a training set. Because of the irregularity of English spelling, the task is far from trivial. The NETtalk experiment opens the door to a host of questions, such as: What kind of network architecture is really suited to problems of this nature? What learning strategies could be used? In particular, is it possible to devise a system based on distributed representations that will be able to not only form abstractions of regularities in the training set but also translate these to other test data to show equally good generalization?

For these reasons, and in order to show that SDM may be used to solve an interesting real-world problem, we have designed an experimental SDM-based system for mapping text to phonemes and have simulated the system on a multiprocessor computer and on a Sun workstation. In the initial version, an iterative supervised learning strategy is employed. A *training set* of English text is presented to the system by means of a moving window containing a fixed number of characters (seven in the current work). As the window progresses through the text, at any given instant the central character in the window is associated with the correct output phoneme. The objective is to "teach" the system to respond with the correct phoneme when prompted by any context of seven consecutive letters of English text. As further training, several more passes are made over the text. Training can be described as a multidimensional search for minimum error on the error surface. The training method is a traversal on the error surface using gradient descent in the direction that reduces the error between the actual and desired output. After completing the training regimen, the system is then tested once again on the training data and scored. Finally, in order to investigate its ability to generalize from learned examples, the system is tested and scored on a different text than the one used for training called the *test set*. The initial results are very encouraging and appear to be in good agreement with results reported by Sejnowski and Rosenberg, 1986.

We will report this work in 1988. We are also planning to implement a text-to-phoneme experiment on either the Connection Machine or on the Stanford prototype.

Bibliography on Visual Encoding Techniques

Bruno Olshausen
(*NCC2-408*)

In order to use sparse distributed memory to perform pattern recognition tasks on visual images, it is necessary that the images be preprocessed in order to reduce the amount of data that must be stored. Sparse distributed memory relies on Hamming distance as a measure of the similarity between bit patterns, and so it is important that visual patterns be encoded such that an object will produce a somewhat constant bit pattern despite changes in viewing perspective. If an image were simply encoded as one bit per pixel, then nearly every different presentation of the same object would produce a different bit pattern entirely unrelated by Hamming distance. This would cause too many different storage locations to be allocated for the same visual object.

It is known that the human visual system does a great deal of preprocessing upon the retinal image in order to extract the most essential information for storage. It is also known that this processing is realized by neurons interacting in a highly organized and massively parallel way. Exactly how the processing is accomplished is not yet known. However, recent advances in neuroscience revealing more of the details of brain architecture and function, combined with the availability of massively parallel computers for simulations, have spurred many efforts at various research centers throughout the world to design artificial-neural-networks for pattern recognition.

The goal of this project is to examine the literature to see whether any of the current neural networks show promise as a technique for preprocessing visual information in an SDM-based system. We have begun an extensive survey of the research literature in the areas of biological and machine vision, neural networks, and pattern recognition, focussing on methods for visual encoding. Our immediate objective is to produce an annotated bibliography and a technical report in mid-1988 that summarizes this information. Based upon this information a simulation will be proposed to investigate the utility of a selected methodolgy for vision encoding in an SDM system.

Speech-Recognition Laboratory

Douglas Danforth
(*NCC2-408*)

The following description concerns a project planned in 1987 for start in 1988.

An autonomous system based on sparse distributed memory must incorporate sensors to observe the world. Acoustic sensors are one example. We anticipate that speech recognition systems, utilizing acoustic sensors, will be important in the command and control of complex and autonomous systems. The purpose of the speech-recognition laboratory is to use problems in speech recognition to learn how to design sensory systems for encoding sensory data, taking advantage of sparse distributed memory's special capabilities for storing and predicting sequences of patterns. The initial objective is to produce a prototype system, using a commercial speech-recognition unit as a front end, that is capable of recognizing thousands of discrete words in real time. The second objective is to use data generated by a digital signal processor to study how SDM may transcribe and interpret continuous speech.

Part of the research concerns discrete speech recognition. Discrete words are utterances that are bounded by silence. For such utterances it is possible to stretch or compress them to a standard length and then convert them to a fixed-length bit pattern. The commercial recognition unit we have chosen uses a recognition algorithm based on Hamming distance between such bit patterns. The sparse distributed memory also uses Hamming distance but there is an important difference. Whereas the commercial recognizer simply uses nearest-neighbor information, SDM uses joint probability distributions in the pattern space. In addition, the massive parallelism of sparse distributed memory permits decision-making in real time. We plan to exploit the speed and correlational information used by sparse distributed memory to build a word-recognition system that is capable of distinguishing thousands of words in real time.

Another part of the research concerns continuous speech transcription. At a low level of processing, speech may be decomposed into a sequence of spectra. Sparse distributed memory provides a natural means of analyzing sequences of such spectra. Given a long sequence of spectra generated from speech in which each linguistic unit (*phone*, *phoneme*, *syllable*, *word*) has been presented in a variety of contexts, the data-pooling mechanism of sparse distributed memory will reinforce the frequently occurring subsequences and suppress the random subsequences. Because the memory is capable of regenerating and predicting stable sequences given an initial segment, the natural sequential clustering of speech data can be used to discover regularities at each level of encoding. For example, if spectra are used then phones can be discovered; if phones are used then phonemes, if phonemes then syllables, if syllables then words. In this way, sparse distributed memory may be used to detect hierarchies in groupings of speech data and to segment the data into linguistic units.

Optoelectronic Cerebellum Network

Egon Loebner (Hewlett-Packard Labs)
James Keeler (RIACS)
David Rogers (RIACS)
Coe Miles-Schlichting (NASA, Code RI)
(*NCC2-408*)

The following is a description of a project planned in 1987 for start in 1988.

Unlike the retinal neural system, which is nearly devoid of memory and learning and which is a feature extracting and information preprocessing network, the cerebellum function is more integrative and correlative. It integrates several sensory modalities and correlates preprogrammed motor responses with the relevant perceptual inputs. Most importantly it adjusts and recalibrates through a multidimensional error-detecting and error-correcting mechanism.

We will attempt to integrate recent findings of Lisberger and his group at The University of California at San Francisco on image slippage during small target pursuit with the latest tools of neural network modelling. From a hardware point of view we will discuss the importance of photochromic and electromic elements to obtain a wide range of adjustable weights at the modelled synapse. From an architecture point of view, we will examine the implications of sparse distributed memory for the functionality of cerebellar and associated systems.

1.4. Telescience

Project Leader: Barry M. Leiner

The telescience and distributed systems area covers a collaboration between RIACS and NASA in studying how to build networks and distributed systems capable of supporting scientific investigations using remote resources. Telescience is NASA's term for the conduct of scientific research using instruments, computers, and researchers separated by wide distances. It includes the remote design and remote installation of experiments on space platforms, the operation of those experiments, and the collaboration of scientists around the world in analyzing and publishing the results.

The goal of the RIACS telescience program is to make telescience a reality in the space station era by advancing, applying, and evaluating the technology base in a widely distributed and coordinated multidisciplinary program. We are accomplishing this through a three-fold thrust. First is the development of advanced information systems to support the various modes of telescience operation. In this area, there are currently two projects; the integration of a Telescience Workstation Environment (being done as part of the Telescience Testbed Pilot Program and collaboratively with the Scientific Computing Environment project area) and the development of advanced networking capabilities (in collaboration with the NASA Science Internet program). The second thrust is the application of technologies to specific scientific disciplines. Currently, we have two projects evaluating the application of telescience to space life sciences in collaboration with the Life Sciences Projects Office. The third thrust is providing assistance in the coordination of multi-disciplinary multi-institutional programs. Here, there are two projects; the Telescience Testbed Pilot Program coordinating a rapid-prototyping testbed environment involving 15 universities, and a project started late in 1987 investigating the institutional and organizational requirements for a program in global environmental research.

Telescience Workstation Environment

Michael A. Slocum
(*Core*)

The Telescience Workstation Environment (TeleWEn) was conceived as an infrastructure to provide a network and workstation model, and a cohesive base workstation environment for telescience users including shared resource management and integration of telescience testbed pilot projects. The project has as its premise that integrating a software environment in a form easily installed and used by the space science community can increase their scientific productivity. The TeleWEn development was broken into two phases, the first being a base level environment, and the second adding shared resource management and integrating testbed pilot projects. Each phase covered both the Sun and DEC Microvax hardware architectures. RIACS took on the responsibility to develop the Sun portion of the environment. Phase one of the environment includes generic user setup files, shell scripts, the Gnu Emacs extendable text editor, the TeX and LaTeX document processing system, the Rand Mh electronic mail handling system, the Utah graphics tool kit, a network monitoring utility, and many miscellaneous or public domain software packages and utilities.

In 1987, phase one of the Sun TeleWEn was completed in December, 1987, and began alpha test at RIACS. In addition to completing the environment, a RIACS software distribution policy was developed.

During 1988, the Phase one environment will be put into service at telescience testbed sites for evaluation and feedback. In addition, integration of Phase two will occur by integrating pilot testbed software and developing shared resource management functionality. Phase two will also explore some of the following areas: the utility of the Mach distributed operating system, the X.11 and Sun NeWS windowing system environments, object oriented programming environment tool kits, advanced workstation and dialogue management systems, interactive help facilities, advanced programming languages and programming environments, multi-media conferencing, and adaptive, distributed resource sharing. The effectiveness of the TeleWEn in improving scientific productivity will be evaluated in the context of the Telescience Testbed through a series of surveys and interviews.

Scientific Networking

Barry M. Leiner
(Core, NCC2-427)

Computer networks are critical to scientific research. They are currently being used by portions of the scientific community to support access to remote resources (such as supercomputers and data at collaborator's sites) and collaborative work through such facilities as electronic mail and shared databases. There is considerable movement in the direction of providing these capabilities to the broad scientific community in a unified manner. In the future, these capabilities will even be required in space, as the Space Station becomes a reality as a scientific research resource. This activity is intended to move the national research network forward through support of the various collaborating agencies.

In 1987, an implementation plan for the Interagency Research Internet was developed and submitted as a white paper to the FCCSET Workshop on Scientific Networking. The Internet Activities Board (IAB) Task Force on Scientific Requirements, chaired by Dr. Leiner, prepared a white paper on networking requirements for scientific research. As the IAB observer to the Federal Research Internet Coordinating Committee (FRICC), support was provided in developing plans for interagency networking.

The focus in 1988 will be to address the critical issues involved in interconnecting networks operated and administered by different organizations in a way that protects the interests of those organizations. To accomplish this, two invitational workshops will be held in coordination with the IAB and sponsored by the FRICC.

Applied Telescience Testbedding

Vicki L. Johnson
(*Core*)

The application of telescience technologies to scientific disciplines and specific experiments is required to objectively measure the effectiveness of telescience principles and research directions. Applied telescience presents several challenges: engineering appropriate telescience tools to enable the discipline science; embedding these telescience technologies in existing or planned space information systems; evaluating the quantitative and qualitative impact of telescience across a wide range of parameters, such as science quality, crew productivity, cost, reliability, and human factors; and finally, where telescience is found to be beneficial, refining science user requirements so that telescience may be incorporated into future information and engineering system flight requirements.

Two rapid prototyping life sciences testbeds are underway at Ames in a joint program between RIACS and the Life Sciences Project Office. The first testbed, designed in the context of Space Lab Life Sciences (SLS-1), focuses on the acquisition of habitat data and teleanalysis of this data from distributed sites. Different approaches to data routing will be evaluated for performance, reliability and ease of operations and maintenance.

The second testbed investigates how telescience technologies might help enable life sciences on the Space Station. Teleoperations experiments will be performed using a small robot arm assistant situated inside a glovebox mockup (multi-purpose workstation). The testbed architecture will simulate a subset of the functions of several elements of the the Space Station Information System. The testbed will be evaluated by performing four common life science scenarios, varying operational modes and telescience technologies, and then comparing effectiveness measures such as productivity and reliability. The scenarios vary the degree of manual versus automated operation and remote versus local control.

In 1987, discipline oriented working groups at Ames were formed to determine common telescience needs. applications and technologies. Funding was secured for the implementation of two ARC life science testbed activities and associated research in applied telescience and telescience infrastructure. One testbed is focussed on upgrading Spacelab capabilities and the other on defining requirements and approaches to life science experiments on board Space Station. RIACS helped design the proposed testbeds and identify participants and needed facilities. Collaboration between the Telescience Testbed Pilot Program and Ames Life Sciences was initiated, particularly between ARC and Stanford University and University of Colorado.

In mid-1988, the life sciences testbeds will be evaluated by discipline scientists performing representative experimental scenarios. The infrastructure technologies and evaluation methodologies may then serve as the foundation for later phases of higher fidelity life sciences testbedding, and/or testbed efforts which encompass other disciplines areas at ARC (e.g. AI and space data systems). Additional ARC testbed participants will be sought, testbed connectivity will be extended from the testbeds to other NASA centers, and ties to the TTPP community will be strengthened through the exchange of research results and technology development.

Telescience Testbed Pilot Program

Barry M. Leiner
Richard Haines
Michael Slocum
Maria Gallagher
(*NASW-4234*)

Space Station and its associated laboratories, coupled with the availability of new computing and communications technologies, have the potential of significantly enhancing scientific research. To assure that this potential is met, scientists and managers associated with the Space Station project must gain significant experience with the use of these technologies for scientific research, and this experience must be fed into the development process for Space Station.

The NASA's Office of Space Science and Applications (OSSA) has initiated a pilot program to validate the user-oriented rapid-prototyping testbed approach. Fifteen universities, under subcontract to the Universities Space Research Association (USRA), are conducting a variety of scientific experiments emulative of the scientific research of the Space Station era and aimed at resolving critical issues in Space Station Information Systems design. The goal is to allow scientists to interact with potential space station technologies in a manner that will allow resolution of design and specification questions without having to wait until space station hardware is available.

During 1987, the fifteen university participants in the program were selected and subcontracts put in place. The selected activities spanned four space science disciplines (Earth Sciences, Life Sciences, Astronomy and Astrophysics, and Microgravity Sciences) as well as a number of technology based investigations with application to several disciplines. Through a series of meetings, both before the contract and since, these activities were coordinated and focussed on addressing a number of well-identified critical issues in the design and development of the information system for the Space Station era. RIACS contributions were in two areas. Technical efforts focussed on the integration and deployment of the telescience workstation environment (see above.) Programmatic efforts focussed on the coordination of the activities to assure the critical issues were addressed effectively. To support this, RIACS put in place facilities such as electronic mailing lists, collated monthly and quarterly reports, and organized exchange meetings. An interim report was prepared documenting initial findings and recommendations.

In 1988, the results of the testbed activities will be integrated into a set of recommendations regarding programmatic approaches to rapid-prototyping testbed activities and the technical requirements for the information system to support space station era science. In addition, the TeleWEn will be used to evaluate the effectiveness of an integrated workstation environment in improving scientific productivity.

1.5. Scientific Computing Environments

Project Leader: Robert L. Brown

The Scientific Computing Environments project exists to develop a complete computational environment for support of all stages of the process of scientific problem solving -- formulating problem statements, abstract algorithms, concrete algorithms, machine codes, executing those codes, analyzing the results and publishing conclusions. The project is a collaboration with branches inside NASA Ames and with the CASIS (Computer Applications in Space Information Systems) project at Stanford University, and includes a testbedding laboratory where experimentation with graphics activities takes place. Additionally, the SCE project governs the RIACS computing facility, which it uses as a "beta" test site for its developments.

Distributed Program Composition System

Robert L. Brown
(*Core*)

A prototype program composition was constructed in 1987. This system, named DPCS, allows the construction of systems of computer programs into larger applications and uses a message-passing, pipe-constructed paradigm. The current prototype can be used to create composed programs with arbitrary (within the operating system limits) interconnection structure. DPCS is oriented towards applications that have large subparts; it is not efficient for constructing large systems from very small parts such as procedures. DPCS uses a graphical interface; the user constructs the program graphically. Icons represent program parts and lines represent communication paths.

The existing implementation does not accommodate the data translations necessary to allow the system to create composed programs to span machine boundaries and automatically handle the data type translations. This will be completed in 1988.

Distributed Concurrent C Debugger

Bill Lynch
Robert L. Brown
(*Core*)

The RIACS concurrent C debugger, CDB, developed in 1986, was extended so that it can be used on a machine other than the one running the concurrent program. We used a 16 processor Sequent Balance system for the concurrent programs and a Sun Microsystems 3/160 for the debugger. When running in distributed mode, event packets are passed across a network to the debugging host. The debugger passes control information (for enabling/disabling event points and break points) back to the concurrent program host. A technical report is in preparation.

Computational Chemistry Graphics

Julian Gomez
(*NCC2-397*)

This task was an initial demonstration of the application of advanced computer graphics to problems in computational chemistry. The result of this task was a videotape showing graphical representations of an O_2+N reaction produced through computational methods and software associated with producing the videotape. Harry Partridge and Rich Jaffe of ARC RTC produced datafiles containing electron potential values in three space evaluated as a function on three parameters over time, and atomic center positions over time. For the first case, a surface was fitted to the points and tessellated with bilinear surface patches (polygons). In the second case, spheres defined by polygons were positioned according the datafile and colored according to role. After these conversions to produce geometry data, animations were designed to show visually what happens during the reaction. The central part of each image focused on the potential surface; the atomic positions were shown through an inset in the corner of the screen. The animations were designed to best portray the surfaces and atoms as the simulation progressed through its duration. The final portion of the videotape is from the viewpoint of an electron potential trajectory as it moves through the reaction. The software used to make this videotape was designed as a general case so it can be used in the future for other computational reactions.

Digital Video Generation

Robert L. Brown
Philip Klimbal
(*Core*)

In the course of generating the computational chemistry trimer atomic system movie, the Scientific Computing Environment project developed a program and methodology that allows the generation of digitized composite video from the traditional red, green, and blue rasters. This allows the creation of graphics in virtual frame buffers, which can then be converted to digitized composite video and very quickly placed on the video editing disk available in the NAS Facility. This is advantageous because, previously, the only way to place data on the video editing disk, an expensive critical resource, was to render the images on a relatively slow Silicon Graphics IRIS workstation. Using that technique, it could take 30 seconds or more to copy one thirtieth of a second of video to the editing disk. The new technique allows ten seconds of video to be placed on the disk in a couple of minutes. The amount of computation required does not drop but now that computation can take place anywhere computation is available. For our movie, we typically simultaneously used 12 of the 16 processors in our Sequent Balance system. The technique of generating the digital composite video allows the creation of graphics far beyond what can be reasonably generated by graphics workstations, and so has captured the interest of the workstation applications branch associated with the NAS Facility.

Facility Upgrades

Philip Klimbal
(*Core*)

Six new SUN 3 workstations were added to the network in 1987. Two MAC-II workstations were also added, one with a color display. One and one half Gigabytes of disk space was added to the network. A new section of ethernet was installed and repeater-linked to the existing segment. Operating system upgrades were performed on the Sequent Balance 21000, the Intel Hypercube, SGI Iris, and SUN systems. The TeX word processing system was installed on the SUN systems. An Appletalk network using PhoneNet was installed and gatewayed to the RIACS subnet with a Kenetics FastPath box. Expansion memory from Helios Systems was added to 6 of the sun workstations. The Utah Raster Toolkit was installed on the Sequent, Sun and SGI systems. An image scanner from Princeton Graphics Systems was added to one of the MAC-II systems.

1.6. Applied Computer Science

Since its inception, RIACS has recognized that the needs of NASA include both basic advances in computer science as well as application of expert skills in computer science and technology to specific NASA programs and projects. The research work in Applied Computer Science provides for close collaboration between RIACS experts in computer science and specific NASA research individuals or groups, and it accommodates several small one-investigator projects that do not fit within the other principal project areas. Although the proposed studies and analyses emphasize the sophisticated *application* of computer science over the discovery of new advances, some fundamental research may be required to provide the expert assistance needed.

NAS Extended Operating Configuration

Eugene Levin
(*NCC2-431*)

During the first half of 1987, Eugene Levin served as a member of an advanced system planning team to establish the requirements, goals and principal technical characteristics of the Extended Operating Configuration (EOC) of the Numerical Aerodynamic Simulator (NAS). He presented a paper entitled "Scaling of Data Communications for an Advanced Supercomputer Network" at the Third International Conference on Data Communications Systems and their Performance in Rio de Janeiro (June 22-25). The paper was published in the Proceedings of that conference.

During the second half of 1987, Levin supported the studies of the advanced NAS EOC by collaborating in joint NAS/RIACS efforts to investigate the potential role of mini-supercomputers. He arranged for the loan of a Convex C1 and an Alliant FX/8 to the Ames Research Center. Processors of this type may be of future importance in advanced NAS configurations. Useful hands-on experience was obtained by performing experiments with these test machines. Of particular importance was the investigation (conducted largely by Convex) of interfacing a Convex to the high speed channel of the Cray 2.

The studies of the NAS Extended Operating Configuration were supplemented in the second half of 1987 by investigations by Ari Ollikainen on advanced networking concepts.

Computational Chemistry

Eugene Levin
(*NCC2-431*)

A major milestone was reached in a study being conducted by Levin together with Harry Partridge and FIRSTNAME? Stallcop of NASA. The high temperature transport properties (such as viscosity, thermal conductivity, etc.) of the major constituents of air (Oxygen, Nitrogen) were correctly determined for the first time. The results of prior *ab initio* computer solutions of the Schroedinger equation were combined with the best available experimental data to obtain complete interaction potentials for both neutral and ion-atom collision partners. These potentials were then used in a computer program developed by Levin to evaluate the collision cross-sections from which the transport properties could be determined. The results are contained in a paper "High Temperature Transport Properties of Air" by Levin, Partridge and Stallcop, which was presented by Levin at the Twenty-Second AIAA Thermophysics Conference in Honolulu (June 8-10). These results are of importance for the Advanced Orbital Test Vehicle (AOTV) and the National AeroSpace Plane (NASP).

The work in computational chemistry with Partridge and Stallcop was continued by further investigations of the transport properties of air. Specific uncertainties relative to charge exchange results were resolved and the computation of collision integrals for the interaction of nitrogen atoms with oxygen atoms was initiated. A further paper on the transport properties of air (to include these later results) was submitted to and accepted by the American Institute for Aeronautics and Astronautics for presentation at the Thermophysics Conference in June of 1988.

Networking Technology

Ari Ollikainen
(*Core*)

Activities in support of the Networking Technology task for the Numerical Aerodynamic Simulation (NAS) Systems Division have consisted in 1987 mainly of establishing membership, participation and contribution in key technical committees/groups involved in computer networking:

1. NAS representative to the NSF funded Bay Area Regional Research Network. Member of Management and Technical Committees.
2. Internet Advisory Board's Internet Architecture Task Force Member.
3. Internet Advisory Board's Scientific Requirements for Networking Task Force Member.

4. NSF Division of Networking, Communications Research and Infrastructure Program Advisory Committee. Member and active participant in Technical Sub-committee.

5. NAS representative to the Federation of American Research Networks.

Work on the definition of the functions of an "Advanced Gateway" with a minimum of 10,000 packets/second throughput has continued. Contacts and visits with potential sources of hardware/software bases were conducted. Several papers are planned to describe the functions of the "NAS Network Server" and a proposed architecture.

Considerable time has been spent in an advisory capacity to the NAS Long Haul Communications group in resolving problems related to the architecture of the Network, NASnet. This will continue as the experiments are performed, analyzed, and integrated into NASnet and also into the Network Server effort.

Paticipation in the NASA Science Internet Project at Ames in an interface role for NAS has expanded into an advisory function. This has included the formulation of a distributed monitoring study activity under Telescience and the formation of a NASA Science Internet Working Group for protocol gateways and ISO protocol rationalization.

Network Analysis

Marjory J. Johnson
(Core, NCC2-480)

The purpose of this project is to analyze high-performance networking and real-time networking in support of the Space Station. Areas of emphasis are the FDDI (Fiber Distributed Data Interface) high-bandwidth token-ring protocol, network protocols to support real-time applications, and network management.

FDDI is an emerging American National Standards Institute (ANSI) and International Standards Organization (ISO) standard for a 100 megabit-per-second fiber-optic token ring. The principal investigator has been conducting performance analyses of FDDI for several years. Two of these analyses were published in IEEE journals this year (see the publication list). FDDI provides support for two classes of service: synchronous service, to support applications which require deterministic access to the channel, and asynchronous service, to support applications which do not have such stringent response-time requirements. During 1987 a study was completed to determine how to set ring parameters to support synchronous traffic most efficiently for Space Station applications. This study is reported in a paper entitled, "Analysis of FDDI Synchronous Traffic Delays," which was accepted for presentation at the IEEE Systems Design & Networks Conference, to be held in April, 1988.

The principal investigator has also participated for several years as an observer on the ANSI committee that is developing the FDDI protocol, to represent NASA's interests. In January of 1987 she made a presentation entitled "FDDI Analysis at NASA Ames" at one of the committee's working-group meetings. As a result, several vendors are seeking NASA input as they design FDDI products.

A major accomplishment during 1987 was the initiation of a collaborative effort between various governmental agencies to evaluate network architectures and protocols with respect to support for real-time applications. In January, 1987 RIACS, in conjunction with the Information Sciences Division (RI), sponsored a workshop at NASA Ames to compare and contrast the FDDI and the Society for Automative Engineers' High Speed Ring Bus (SAE AE-9B HSRB) token-ring protocols and to discuss possible approaches to designing network protocols to support real-time applications. There were forty-five attendees at the workshop, representing NASA, DoD, defense contractors, vendors, and independent consultants. Presentations at the workshop included discussions of end-user requirements, design philosophies of FDDI and HSRB, and analyses of performance of the two protocols. A summary of presentations made at the workshop, along with an independent comparison of the two token-ring protocols, is contained in a RIACS technical report, "Network Protocols for Real-Time Applications," RIACS TR 87.15, May, 1987. At the end of the workshop a strong commitment was expressed within the group to continue joint discussions to determine a suitable network architecture to support real-time applications.

A follow-up workshop was held at the National Bureau of Standards in June, 1987, to discuss communication requirements for real-time systems; a workshop to discuss experiences with existing real-time communication systems is scheduled to be held at NASA JSC in January, 1988. The group has become international in scope, representing the broad interest in networking for real-time applications. Discussions to determine how best to coordinate research efforts in real-time protocols are continuing between RIACS and researchers both at the Swedish Institute for Computer Science (SICS) and at the French Institut National de Recherche en Informatique et en Automatique (INRIA).

The final area of emphasis in this project is network management. The principal investigator has been following the substantial

activities that are currently underway in various standards bodies in this area. In October, 1987, she participated in a "National Bureau of Standards (NBS)/MITRE Workshop on Network Management Functional Requirements." Topics discussed at this workshop, which was conducted to further the standards efforts, included both issues that are being addressed within the standards bodies and issues that are not, such as the application of expert systems to network management, coordination between multiple managers, and efficiency of network management operations. Also, in 1987, a collaborative effort was initiated with the National Bureau of Standards to study network-management protocols. NBS, in conjunction with the MITRE Corporation, is establishing a testbed which will be used to conduct these studies.

Plans for 1988 are to direct network analysis more specifically to the Space Station Information System (SSIS), which encompasses the local area network onboard the Space Station, space-to-ground links, and wide-area ground networks.

NASA Computer Science Research Program Plan

Michael R. Raugh
Marjory J. Johnson
(*Core*)

The computer science research program within NASA, established in FY 83, is managed within the Office of Aeronautics and Space Technology (OAST). A "NASA Computer Science Research Program Plan," (NASA Technical Memorandum 85631), was published in March, 1983 to set forth the goals of the program, to describe an approach to be taken to create and expand the program, and to give a program overview. In November of 1986 an OAST Intercenter Planning Committee was formed to review the status of the NASA computer science research program, and to update the program plan. The program currently consists of four areas: advanced computer system architecture, software engineering, scientific and engineering information management research, and artificial intelligence. During 1987 Raugh and Johnson coordinated activities within the Intercenter Planning Committee in the area of scientific and engineering information management research. They integrated input from representatives of all the NASA centers concerning the state of the art, technology needs, and current NASA activities in this area. During April of 1987, Intercenter Planning Committee members made an oral presentation on the NASA Computer Science Research Program to a review panel consisting of members of the National Research Council. Raugh presented the information-management area at this review. During the remainder of 1987 Raugh and Johnson worked jointly to prepare the information-management portions of a written NASA Computer Science Research Program Plan. Major sections of this plan include a presentation of goals and objectives of the plan, a discussion of NASA's computational requirements, a description of the current program and accomplishments during the first five years of its existence, and a discussion of directions for future expansion. This document is intended to serve as a guide for computer science research within NASA for the next five years. It was near completion at the end of 1987; it will be published as a NASA Technical Memorandum in 1988.

The Illinois Computer Laboratory for Aerospace Systems and Software (ICLASS) is funded by a block grant from NASA as part of its Computer Science Research Program. In November, 1987, Johnson attended a review of the activity being conducted under this grant.

Planning and Learning Research

Peter Cheeseman
Monty Zweben
(*NCC2-428*)

Autoclass is a program for automatically classifying records of a database into the most probable classes. The program examines the database and determines which classes are present, then it computes the probability that each record belongs to each class. This program was applied to the database from the infrared astronomy (IRAS) space telescope and produced several new discoveries. Owing to the successes of the automatic classification project, and requests for extra effort in this direction from NASA, nearly all work in the second half of 1987 and beyond has been concentated on implementation, testing and development of the AutoClass program.

The Autoclass Project is supported by the AI Research and Applications branch (RIA) of the AMES Information Sciences Division. Unlike previous automatic classification programs, the Autoclass system does not need to be told how many classes are present or even if there are any classes at all. Autoclass uses a new extended Bayesian approach that searches for the most probable classification, and assures that any classes that are found have a real cause---i.e. classes are not an artifact of the search process. The current status of the class induction program (Autoclass II) developed under this project is:

1. A version of Autoclass II has been written and extensively tested on several large data bases. The results of these tests have been more successful that originally envisioned, as described below.
2. One bad feature of the first version of AutoClass II was a tendency to overclassify; as revealed by testing on both real and test data. This tendency was a result of a conceptual error in the test for the number of classes. An approximate test based on the Bayesian approach has now been implemented. and shown experimentally to fix the problem.
3. A proof has now been discovered that shows that Autoclass II is performing a maximum posterior probability search, and not just an approximation as originally thought.

As part of the verification process, AUTOCLASS was applied to IRAS satellite spectral data provided by the Space Science Division's Astrophysics Experiments branch (SSA). The data consists of 5,543 spectra that make up the low resolution spectral data base. Autoclass produced 78 classes in this set, and many of these have also been futher split into subclasses. The results of these investigations as being examined by astronomers.

Several of these classes are of real astronomical significance. They could not have been found by the standard statistical analysis used to reduce the IRAS data, as that method relied completely on previously known IR spectral classes. Essentially, the method employed was to describe the features of known IR objects, and then simply to determine which class a given spectrum most closely resembled. The major contribution of AUTOCLASS is that it makes no apriori assumptions, and is thus able to discover previously unacknowledged object types. These results are an exciting step in the direction of fully automated data analysis, and it is expected that more data and improved versions of AUTOCLASS will lead to many unexpected discoveries. The theory and results have been described in several publications, with several more in preparation. In addition, the work has been described in a number of presentations across the country.

In 1988, the work will continue. The current AutoClass program is the result of applying Bayesian inductive methods to the simplest type of pattern---automatic class discovery. Even within this type of model we chose the simplest assumptions imaginable in order to get started. Given the simplicity of the model, it is surprising how well it worked; however testing has revealed weaknesses due to the simple assumptions. These assumptions can be relaxed, and much work in the immediate future is directed to this end. In particular, the following extensions are planned.

1. Relevent/Irrelevent Attributes. That is, allowing attributes to be relevent or irrelevent for individual classes. Currently, AutoClass assumes every attribute is relevent to distinguishing every class. This is a good assumption for uniform data, such as the IRAS spectral data, but it is not true in general. This extension allows the parameters associated with different classes to be equated, so that their values are not assessed separately. This will allow much finer class distinctions, especially when the classes are very similar but differ on a few attributes only.
2. Dependency Models. Instead of assuming conditional independence of attributes within each class, this extension will allow various kinds of dependency. Simple covariance matix methods are currently being explored,
3. Multiple Classifications. Instead of assuming that the classes are mutually exclusive and exhaustive, it is possible to build models with overlapping ("independent") classifications.
4. Directed Models. Instead of discovering intrinsic classes in a given data set, it is often desirable to discover classes that are maximally predictive of a given attribute (the target attribute). Several models for doing this are being explored.

In addition to the above extensions to classification models, the Bayesian approach can be extended to other models, such as time series analysis, spectral analysis, failure analysis etc. Preliminary investigations on temporal models has been promising so far.

The Science of Computing

Peter Denning
(*Core*)

This project is an ongoing effort to identify connections between the fundamentals of computer science and other scientific disciplines, mathematics, and engineering. The results have been published as a series of RIACS technical reports and articles in *American Scientist* magazine. The topics covered in 1985 were networks, parallel computing and its evolution, and the arbitration problem that arises when a circuit must distinguish between near simultaneous events. In 1986 they were expert systems, intelligent machines, random access memory, virtual memory, amd electronic publishing.

Six topics were studied in 1987. *Security of Data in Networks* [RIACS TR-86.26] examines the problem of authentication of users, resources, and transmissions in a large network, such as NASA envisages with Telescience; cryptography is an essential tool. *Evaluating Supercomputers* [RIACS TR-87.2] examines the questions that must be faced in developing a theory of performance evaluation for supercomputers of parallel architecture; the current theory is not powerful enough for this case and does not include measurement of the usability of systems. *Multigrids and Hypercubes* [RIACS TR-87.8] examines the mapping of multigrid solutions to elliptic partial differential equations to hypercube architectures; when binary reflected Gray codes are used to associate nodes of the grid with nodes of the computer, muligrid algorithms are optimal. *Computer Modeling of AIDS Epidemiology* [RIACS TR-87.14] examines how large computers can solve differential equations representing the spread of a disease and enable policy planners to assess the effects of possible controls. *Baffling Big Brother* [RIACS TR-87.19] returns to the question of cryptographic protocols for authentication in large networks,

here considering protocols that provide anonymity for the initiator of a transaction while continuing to assure the recipient that the intended result will occur. *A New Paradigm for Science* [RIACS TR-87.24] examines the impending breakdown in scientific research that will arise out of the "information explosion" and advocates new approaches to collecting, classifying, and disseminating scientific results.

Candidates for study in 1988 include a review of deadlock theory as it applies to parallel programming (deadlock occurs in a multiprocess system when a set of two or more processes are holding resources and are stopped waiting for others in the set to release resources); an analysis of why designers of intelligent systems have so much trouble with the rule-based approach; an analysis of the threat posed to networks of computers by computer viruses; and speeding up parallel computations.

1.7. Software Systems

We have developed software in our facility that may be useful in other research facilities and are available on a no-support basis to groups interested in collaborating on the research:

1. The Concurrent C compiler, runtime system, and debugger are available.
2. Demonstrators for Sparse, Distributed Memory on the Sun workstations are available.
3. Graphical control panels for remote computations initiated from a Sun workstation are available; one of them, called functionview, controls computations whose results are displayed as graphs.
4. A family of software making mouse, pop-up menus, and other aspects of modern bitmapped graphics is available to users of Unipress emacs in UNIX with BitGraph terminals. A BitGraph emulator for Sun workstations is available.
5. UNIX tools have been integrated into the Silicon Graphics IRIS workstation environment and are available.

2. ADMINISTRATIVE REPORT

2.1. Personnel

2.1.1. Staff

In 1987, the research staff were grouped into five project areas: *advanced algorithms and architectures*, which deals with the performance of highly parallel architectures and algorithms for using them; *sparse distributed memory*, which deals with a highly parallel architecture for pattern computation and properties like those of human memory; *telescience*, which deals with the conduct of science via networks; *scientific computing environments*, which deals with a workstation and graphics environment to support the routine parts of the processes of scientific investigation; and *applied computer science*, which deals with other major investigations RIACS has undertaken for NASA that do not fall in the previous categories.

Table 4 shows the staff size by category since RIACS began operations in 1983. During 1987 38 persons were employed by the institute: 14 scientists, 6 research associates, 6 visiting scientists, 7 support staff, and 5 students.

The institute's funds come from a variety of sources. About $1M a came from NASA HQ and is usually called the "core funding." About $1.5M more came from a variety of supplements from Ames divisions. Many of these supplements were added to the core cooperative agreement (NCC2-387) rather than being issued as separate cooperative agreements. Although the funding can be divided into categories such as core and supplement, the personnel cannot be so neatly classified once the supplements are added to the core agreement. For this reason, we have not tried to maintain a distinction between "core" scientists and "other" scientists.

TABLE 4
Staff Sizes

Category	FY 83	FY 84	FY 85	FY 86	FY 87
Scientist	3	9	15	12	14
Research Assoc/Asst	-	-	-	2	6
Visitors	1	3	4	3	6
Summer Students	-	2	5	4	4
Other Students	-	-	-	3	1
Support	1	3	5	6	7
TOTALS:	5	17	29	30	38

2.1.2. Employment Grades

In 1987 we used this set of position titles:

1. DIRECTOR.
2. SCIENTIST. Appointment is from one to three years and requires a PhD or equivalent experience. Senior scientists with outstanding reputations can receive 5-year appointments.
3. RESEARCH ASSOCIATE. An appointment of one to three years for a professional with Master's Degree or equivalent engaged in systems development or applications support for scientific projects.
4. RESEARCH ASSISTANT An appointment of one to three years for a professional with Bachelor's Degree or equivalent engaged in systems development or applications support for scientific projects. This person is closely supervised by a Scientist.
5. SUPPORT. A member of the administrative, clerical, or facilities staff.

All appointments are renewable and subject to continuation of funding.

We drafted a new set of positions, arranged in a ladder that would allow research scientists to ascend to the same salary levels as managers. The initial version of the ladder was modeled after the step system for faculty in the University of California. We will continue discussion of the ladder in 1988 and reach a conclusion by mid 1988.

2.1.3. Management Committee

In August 1987 the Director designated five people as Project Leaders and created a management committee consisting of these persons:

Director	Peter Denning
Deputy Director & Project Leader for Applied Computer Science	Gene Levin
Chief Scientist & Project Leader for Sparse Distributed Memory and for Advanced Algorithms & Architectures	Michael Raugh
Senior Staff Scientist & Project Leader for Telescience	Barry Leiner
Project Leader for Scientific Computing Environments	Bob Brown

In December 1987, the Director retained International Consulting Services, Inc., to advise and work with us on transforming our organizational structure to adapt to the growth we experienced in 1987 and to accommodate the sharp additional growth expected in 1988. This process was expected to lead to a reorganization by mid 1988.

2.2. Computing Facilities

The RIACS Computing Facility Project is managed by the facility manager (Bob Brown) with advice from the Facility Committee. The project seeks to provide computing services that support the research missions of the institute through advanced workstations and networking. Because of the diversity of needs of the various projects, we have based our facility on a resources model that separates computation, storage, and interface and allows for different machines of the network to provide each function. The distinction between computation and storage is familiar to most of us, and protocols have been devised to support it -- e.g., file servers, network file systems. The distinction between computation and interface is less familiar and we normally think that the user interface is part of the workstation. In recent years, however, protocols have been devised to support a separation between compute servers and interface servers -- e.g., the X-window protocol of MIT's Project Athena. The computation-interface distinction makes possible new, cost effective approaches to providing service -- computation can be performed on a shared multiprocessor and individualized window service can be performed by a low-cost X-window workstation.

Our computing facility operates against this background with these goals:

1. Each researcher should have a workstation or window server providing multiwindow and graphical display services, local computing, access to file servers, and access to other computers via network.
2. The facility should use standards that promote vendor independence and accommodate an open system philosophy -- e.g., UNIX, TCP/IP, Ethernet, PostScript, X-windows. These standards allow for the attachment of special equipment and for installation of special software packages.
3. Large-scale computing will be obtained from Ames computational facilities as needed.
4. Access to the internet will be provided, as well as connectivity with Ames research groups involved in joint projects with the institute.
5. The facility should promote the daily use of parallel and distributed computing.
6. Three classes of use should be explicitly accommodated: a) Basic use provides interfaces to applications only, b) Research Development use provides a transportable programming interface, and c) Systems Development provides full capabilities for system designers beyond those provided for (a) and (b).

In support of these goals in 1987, we used our capital equipment fund to purchase workstations, upgrade the network, and add computing and storage capacity to accommodate the growth of the staff. The X-window protocols were still in an experimental stage; their installation will come in mid 1988. At the end of 1987, the following equipment were connected to the ethernet:

Sequent Balance 21000 (16 processors, 16 Mbytes memory)
VAX 11/730 network gateway
11 Sun-3 workstations with file server
Intel iPSC hypercube (32 processors)
Silicon Graphics IRIS 2500 graphics workstation
Evans & Sutherland PS 350 graphics workstation
3 Macintosh workstations

We opened discussions for additional equipment in 1988 to support the missions of RIACS:

1. An Encore Multimax multiprocessor with the Mach operating system as a beta-test.
2. A fully configured Stellar superworkstation with 3D graphics and animation systems.
3. A fully configured Ardent superworkstation with 3D graphics systems.
4. Window servers to support the X-window protocol.

2.3. Supplemental Agreements

Our core agreement with NASA authorizes RIACS to enter into special collaborations funded separately by Ames research groups. The number and level of these supplemental agreements was much higher in 1987 than in previous years. Table 5 is a summary of all the supplemental agreements. Table 6 is a summary of the number and level of agreements by Ames Division.

TABLE 5
Supplemental Agreements Summary 1987

Code	Name	Total of Budgets	1987 Share
DS	Space Station & Advanced Technology Office	$763,700	$177,200
FL	Aerospace Human Factors Research Division	1,107,400	604,000
FSI	Simulation Investigations Branch	42,700	21,400
FSN	Aircraft Guidance and Navigation Branch	11,500	7,700
RI	Information Sciences Division	50,200	23,200
RIA	Artificial Intelligence Research Branch	286,200	143,100
RII	Intelligent Systems Technology Branch	320,100	183,700
RND	NAS Systems Development Branch	103,600	41,400
RNS	NAS Computational Services Branch	168,600	84,300
RTC	Computational Chemistry Branch	177,600	82,000
TOTAL			$1,518,000

TABLE 6

Supplemental Agreements 1987

GRANT #	JOB CODE	PROJECT TITLE	TECHNICAL MONITOR	NASA CODE	PRINCIPAL INVESTIGATOR	START DATE	END DATE	BUDGET
NASW-4234	800	Telescience Testbed Pilot Program	Ervin Schmerling	DS	Barry L Leiner	05/01/97	12/31/88	351,006.00
NCC2-387	870	Research Institute for Advanced Computer Sci	Marcelline Smith	RC	Peter Denning	01/01/96	12/31/88	2,992,887.00
NCC2-387	870-CS3	Network Analysis	Terry Grant	RII	Marjory Johnson	11/01/97	12/31/88	120,316.00
NCC2-387	870-TS1	Space Station Life Science	Darryl Rasmussen	DS	Barry Leiner	12/01/87	12/31/88	90,913.00
NCC2-387	870-TS2	Space Lab Life Science	Frank King	DS	Barry Leiner	12/01/87	12/31/88	56,081.00
NCC2-387	870-TS3	Global Environment	WE Berry	DS	Richard Johnson	11/16/87	05/31/88	77,756.00
NCC2-387	870-TS4	Distributed Monitoring	W.P. Jones	DS	Barry Leiner	12/11/87	12/31/88	187,954.00
NCC2-397	397	NAS Advanced Graphics	Larry Hoffman	RTC	Julian Gomez	01/01/86	02/28/88	177,563.00
NCC2-398	398	Improving Computer Security	William Kramer	RHS	Matthew Bishop	10/01/85	09/30/88	168,648.00
NCC2-399	399	Data Networks Concepts	Terry Grant	RII	Marjory Johnson	11/01/85	10/31/87	99,807.00
NCC2-402	402	Scientific Libraries for the CRAY-2	JT Barton	RND	Don Calahan	10/01/96	03/31/89	103,600.00
NCC2-408	408	Sparse Distributed Memory	Michael Shafto	FL	Pentti Kanerva	01/01/86	10/31/88	1,107,423.00
NCC2-427	427	Space Station Telescience Concepts	Henry Lum	RI	Barry Leiner	06/01/86	06/30/87	50,151.00
NCC2-428	428	Planning and Learning	Peter Friedland	RIA	Peter Cheeseman	05/01/86	04/30/88	286,153.00
NCC2-431	431	Analysis of NAS,EOC, and Computational Chemestry	Ron Bailey	RN	Eugene Levin	06/01/86	02/29/88	250,140.00
NCC2-435	435	Computer Security Analysis	Chester Jew	FSI	J. Michael Long	12/01/86	01/31/87	42,719.00
NCC2-448	448	Air Traffic Control Advisor Expert Systems	Leonard Tobias	FSN	Ronald Chrisley	10/01/86	06/30/87	11,491.00
NCC2-480	480	Management of Distributed Systems	Terry Grant	RII	Marjory Johnson	11/01/86	10/31/87	99,992.00

3. PERSONNEL REPORT

PETER J. DENNING, Director (1983)

Denning came to RIACS in June, 1983, from Purdue University, where he was Head of the Computer Sciences Department since July 1979, a Professor of Computer Sciences since 1975, and an Associate Professor of Computer Sciences since 1972. He spent four years as Assistant Professor of Electrical Engineering at Princeton University after receiving his PhD from MIT in 1968. He received an MS degree from MIT in 1965 and a BEE degree from Manhattan College in 1964. His research interests have included operating systems, computer systems architecture, parallel systems, networks, and analytic performance modeling. He has written over 165 technical articles, 100 technical reports, and two books in these areas since 1967. He served as President of the Association for Computing Machinery (ACM) 1980-82 and Vice President 1978-80. He was Editor-in-Chief of the ACM Computing Surveys 1977-78; he is the consulting editor for the MIT Press Series in Computer Science; and he has been Editor-in-Chief of the ACM Communications since 1983. He is a Fellow of the IEEE and of the American Association for the Advancement of Science (AAAS). He holds two best-paper awards. He received an honorary Doctor of Laws degree from Concordia University in 1984 and an honorary Doctor of Science degree from Manhattan College in 1985.

GEORGE B. ADAMS III, Research Engineer (1983-1987), Visiting Scientist

(1987-)" Adams joined RIACS in August, 1983 and worked full-time at RIACS until August, 1987. Since then he has been a Visiting Scientist with RIACS and an Assistant Professor of Electrical Engineering with Purdue University. Adams received the BSEE degree from Virginia Polytechnic Institute and State University in 1978, the MSEE degree from Purdue in 1980, and the PhD in electrical engineering from Purdue University in 1984. His research interests include computer architecture, parallel processing, interconnection network design, and parallel processing algorithms. He has published over 20 papers and reports in these areas and holds a patent for the Extra Stage Cube interconnection network. During 1986 he was a Lecturer with the Department of Electrical Engineering, Stanford University, and taught graduate courses on computer design. He is a member of Eta Kappa Nu, Tau Beta Pi, Phi Kappa Phi, and Sigma Xi.

MATT BISHOP, Scientist (1984-87)

Bishop joined the RIACS staff in October, 1984, after completing his PhD in Computer Science at Purdue University (1984), where he also received an M.S. (1981). He received an M.A. in Mathematics (1978) and an A.B. in Astronomy and Applied Mathematics (1976) from the University of California at Berkeley. He spent over a year at the Megatest Corporation as a programmer developing and maintaining UNIX subsystems. His areas of interest are operating systems, computer security (PhD thesis), the application of computers to mathematics, and "user-friendly" systems.

NANCY R. BLACHMAN, Systems Programmer (1985)

Blachman joined RIACS in August, 1985. After completing a B.S. in Mathematics at the University of Birmingham in England (1978) and a M.S. in Operations Research at the University of California at Berkeley (1979), she worked four years for AT&T Bell Laboratories doing systems engineering and software development. She then joined Resonez as a software engineer, where she developed image processing software and system software for a prototype magnetic-resonance imaging system. She has published several articles on UNIX systems administration. At RIACS, Nancy worked with researchers on parallel and distributed processing, while serving as a systems programmer. She is also taking courses toward a M.S. in Computer Science at Stanford University.

ROBERT L. BROWN, Scientist (1983)

Brown came to RIACS in June, 1983, from Purdue University, where he was a Research Assistant in the Department of Computer Sciences. He also served as a Graduate Instructor in charge of a computer science course with an enrollment of over 500. He received his Bachelor's degree in Mathematics from the Ohio Wesleyan University in 1975 and his PhD in Computer Science from Purdue University in 1988. His research areas are operating systems, distributed systems, and programming systems. He has written papers on distributed systems and extensions to the virtual UNIX kernel. His current research involves the specification and design of a graphical distributed programming system for a heterogeneous distributed system. Also, he headed the project to develop and distribute the RIACS Concurrent C compiler for the Sequent multiprocessor.

DONALD A. CALAHAN, Visiting Scientist (1984)

Calahan became a consultant to RIACS in 1984. He is a Professor at the University of Michigan, Ann Arbor, Department of Electrical Engineering and Computer Science. He received his BSEE degree from the University of Notre Dame in Electrical Engineering (1957), his MSEE from the University of Illinois-Urbana in Electrical Engineering in (1958) and his PhD in Electrical Engineering from the University of Illinois-Urbana, (1960). He was a professor of Electrical Engineering at University of Illinois-Urbana from 1960 to 1965. His area of interest is large-scale scientific computation, and he has written three text books in the field of Electrical Engineering. He is a IEEE Fellow and he is cited in Who's Who of America.

TONY F. CHAN, Visiting Scientist (1986-87)

Chan came to RIACS in January of 1986, on sabbatical from Yale University where he was Associate Professor of Computer Science. He received his PhD from Stanford University in 1978, and his M.S. and B.S. degrees from California Institute of Technology in 1973. His research interests include efficient algorithms in large scale scientific computing, parallel algorithms, and computational fluid dynamics. After completing his sabbatical visit to RIACS, he took a position as Professor of Mathematics at UCLA. He continues his association with RIACS by visiting two days monthly.

PETER CHEESEMAN, Scientist (1985)

Cheeseman came to RIACS in April, 1985, from SRI International, where he was a Senior Computer Scientist in both the Robotics and Artificial Intelligence Laboratories. Prior to that he was a Professor at the New South Wales Institute Technology, Sydney, Australia. He received his PhD in 1979 from Monash University in the area of Artificial Intelligence. He received his M.Phil in Applied Mathematics in 1973 from Waikato University (New Zealand). His research interests include Artificial Intelligence and Automatic control, induction of models under uncertainty, Bayesian inference, expert systems and robotics. He has published a number of papers in these areas and organized workshops on these subjects.

RONALD CHRISLEY, Student (1986-87)

Chrisley came to RIACS in October, 1986, to continue developing the ATC Advisor, a time-based air traffic control AI system, in conjunction with the Aircraft Guidance and Navigation Branch at NASA Ames. He began working on the project in June, 1986 in a NASA Ames summer employment program. Before that, from September 1985 to June, 1986, he was a research assistant for Dr. Derek Sleeman of Stanford (now Chairman of the Computing Science Department at the University of Aberdeen), developing PIXIE, an intelligent tutoring system. From March, 1986 to September, 1986 he assisted a psychology doctoral student at Stanford University in evaluating the cognitive applicability of connectionist networks. He intends to graduate with honors in June, 1987 from Stanford University with a BS in Symbolic Systems. Although his concentration in his major is computer-generated sound and music applications of digital signal processing, his honors thesis will center around his work with RIACS: the AI issues of hypothetical planning in the ATC domain. Other interests of his include neural network approaches to pattern recognition.

SAM CIRAULO, Administrator (1984-87)

Ciraulo joined the RIACS staff in October, 1984 after holding a position as Administrator, Telecommunications and Purchasing, at Boole and Babbage, Inc. of Sunnyvale, California. He has also served in an administrative capacity at ROLM Corporation and Dysan Corporation, both of Santa Clara, California.

BARBARA CURLETTE, Resource Analyst (1987)

Curlette joined the RIACS staff in November, 1987 after working for Sierra Scientific as Marketing/Sales and Human Resource Administrator. She attended San Jose State University and received a B.S. in Organizational Behavior from the University of San Francisco in 19 86.

LEO DAGUM, Student (1987)

Dagum received his M.S. from the Department of Aero and Astrophysics at Stanford University in 1987. His research has been concentrated in simulation of fluid flows by particle methods.

DOUGLAS G. DANFORTH, Scientist (1988)

Danforth joined RIACS in January, 1988 to work with the Sparse Distributed Memory (SDM) project after being at Wang Labs for seven years as sectional manager of Signal Processing in their Voice Engineering Department. He received his Ph.D. in Education from Stanford University in 1978. His thesis, entitled "Creative Sequential Classification: an Adaptive Approach to Machine Understanding of Continuous Speech", was on the construction of an automatic speech recognition system that could discover regularities in speech waveforms. He earned a M.S.in Statistics in 1974 from Stanford where, for two years, he was a statistics consultant at The Center for Advanced Study in the Behavioral Sciences. Early work in physics began at Stanford where Danforth earned a B.S. in 1966 and continued at the University of Maryland with a M.S. in 1971. At RIACS, Danforth is working to remove many of the artificial constraints found in the field of speech recognition by applying forms of SDM to front-end encoding, phonetic transcription and grammatical dependencies.

LORRAINE FISHER, Project Secretary (1988)

Fisher joined the RIACS staff in January, 1988. Prior to RIACS she was employed at the AMPEX Corporation for one and one-half years. She is currently attends San Jose State University, where she is working on a B.S. in Computer Science.

MARIA L. GALLAGHER, Telescience Project Coordinator (1986)

Gallagher joined the RIACS staff in April, 1986. Prior to that she had been employed in a variety of administrative positions, as a teacher, and as a computer consultant. She received her A.A. in Political Science from the College of San Mateo, and her B.A. in History and Political Science, as well as a Lifetime Teaching Credential, from San Jose State University. Graduate work on her combined M.A. (Art Education/Child Psychology) and Early Childhood Teaching Credential has been completed at San Jose State University. She has completed additional coursework in Math, Engineering, Networks and Computer Science towards a degree in Computer Science. During 1987, Gallagher acted as Project Coordinator for the Telescience Testbed Pilot Program and the Telescience Project Area, as well as, Research Facilitator for RIACS. Gallagher has also provided the administrative support for Ames-wide personnel involved in the Telescience area.

EROL GELENBE, Visiting Scientist (1987)

Gelenbe received his Ph.D. in Mathematics from the University of Paris in 1973. His current research interests are concentrated in analysis of parallel systems.

JULIAN E. GOMEZ, Scientist (1986-87)

Gomez came to RIACS in January, 1986 from Cranston/Csuri Productions Inc., where he was director of R&D. He received his PhD in Computer and Information Science from Ohio State University in 1985. Prior to his graduate program he spent four years in the Computer Graphics Laboratory at the Jet Propulsion the Jet Propulsion Laboratory. He received an AB with honors from the University of California at Berkeley in 1977. His areas of interest are 3-D computer animation, chaos and natural phenomena and he has published several technical papers in these areas. He is a member of ACM, SIGGRAPH, and the IEEE Computer Society. He is Chairman of the Bay Area ACM/SIGGRAPH Technical Interest Group on Performance Evaluation. He is a member of the ARC Code R Computer Graphics Planning and Review Committee.

KATHRYN HAWKEN-CRAMER, Executive Secretary (1987)

Hawken-Cramer joined the RIACS staff in June, 1987. Prior to that she worked as sales office manager for Robinson-Nugent, an electronic components firm and as hostess/manager at Scott's Seafood Restaurant in Palo Alto. She studied chemistry at the University of Arizona from 1974 - 1977.

LOUIS JAECKEL, Scientist (1987)

Jaeckel came to RIACS in August, 1987, to work with the Sparse Distributed Memory group. He received his B.A. in Mathematics from UCLA in 1961, his M.A. in Mathematics from Harvard University in 1962, and his PhD in Statistics from U.C.-Berkeley in 1969. He was a member of the technical staff at Bell Laboratories from 1970 to 1972, and an assistant professor of statistics at U.C.-Berkeley from 1972 to 1977. His interests include mathematics, statistics, and computer science.

UMESH JOGLEKAR, Research Assistant (1986)

Joglekar came to RIACS in November, 1986. He is currently completing M.S. degree in Computer Science at the University of California, Santa Barbara (expected Summer 1987). His education includes B. Tech. (1971) in Aeronautical Engineering from Indian Institute of Technology, Bombay, and Postgraduate Diploma in Management (1973) from Indian Institute of Management, Calcutta. His interests include Artificial Intelligence, Neural Networks/Connectionist Models and their applications.

During 1986, Joglekar started working on "Application of sparse, distributed memory to text-to-speech conversion."

MARJORY JOHNSON, Scientist (1984)

Johnson came to RIACS in January, 1984 from the University of Missouri at St. Louis where she was an Associate Professor of Computer Science. She received a B.A. Degree in Mathematics from Florence State University and M.S. (1968) and PhD (1970) degrees in Mathematics from the University of Iowa. From 1970-1976 she was an Assistant Professor of Mathematics at the University of South Carolina. From 1976-80 she was a systems analyst at NCR Corporation, where she worked with microcomputer systems and computer communication networks. She has published technical papers in both mathematics and computer science.

RICHARD G. JOHNSON, Visiting Scientist (1988)

Johnson joined RIACS as a visiting scientist in January, 1988. Prior to that he was Acting Science Advisor to the President and Acting Director of the White House Office of Science and Technology Policy (OSTP), Executive Office of the President from May to October 1986. Dr. Johnson was Assistant Director for Space Science and Technology in OSTP from November 1983 to October 1987. He received his B.S. in Physics from Antioch College in 1951 and his Ph.D. in Physics from Indiana University in 1956. That same year, he joined the Lockheed Palo Alto Research Laboratory of the Lockheed Missiles and Space Company, where, for 27 years, he conducted a broad range of research in low energy nuclear physics and in the space sciences. He was Manager of the Space Sciences Laboratory for ten years and Senior Science Advisor to the Director of Research for five years. He was a visiting professor at the University of Bern in 1980.

VICKI JOHNSON, Research Associate (1987)

Johnson joined RIACS in July, 1987. After completing a B.S. in Computer Science at the University of New Mexico (1978), she joined Bell Laboratories as a systems engineer in enhanced network services. While working for Bell, she obtained a M.S. in Computer Engineering from Stanford (1980) and an MBA from NYU (1984). She then joined the Planetary Geosciences Division of the Hawaii Institute of Geophysics as a computer scientist, developing analysis and data systems to support imaging spectroscopy. At RIACS, Vicki is working as a research associate in telescience applications and infrastructure.

PENTTI KANERVA, Scientist (1985)

Kanerva came to RIACS in October of 1985 from Stanford University. While at Stanford, he developed a large terminal network, software and hardware for technical writing, typesetting software, algorithms for searching and sorting, new memory architectures, new CPU architectures, and directed the technical development of a campus-wide network for text-handling. He was also a postdoctoral fellow and is presently a visiting scholar. In 1965-1967 he organized the first computer center for the University of Tampere, Finland, and taught mathematics, statistics, and computer programming. Prior to working at Tempere, he worked in experimental design and statistical analysis at the Forest Research Institute of Finland and the Finnish State Computer Center. He has a PhD in Philosophy from Stanford (1984) and MS in Forestry from the University of Helsinki (1964). His current research is based on his dissertation on memory and is aimed at building practical computers with some of the desirable properties of human memory. His interests include computer-aided communication by people, the computing power of a network of computers, and the design of systems for people without a strong background in computers.

JAMES KEELER, Student (1987)

Keeler received his Ph.D. in Physics from the University of California at San Diego in 1987. His research is concentrated in neural networks.

WILLIAM C. KESSINGER, Student (1987)

Kessinger came to RIACS in June, 1987 while a coterminal B.S./M.S. student in Industrial Engineering at Stanford University. During the summer of 1987 he worked with Marjory Johnson in the simulation and statistical analysis of networks using the FDDI protocol. The primary result of his efforts was a report on the effects of the distribution of asynchronous traffic on message delays. He is presently working part-time at RIACS while finishing his M.S. In August, he will be leaving to take up full-time employment at the Prudential Investment Corporation in New York.

PHIL KLIMBAL, Research Associate (1987)

Klimbal joined the RIACS staff in April, 1987 from the Engineering Computer Network at Purdue University where he was a system programmer working on BSD UNIX kernel enhancements and systems support. He received a B.S. in (Computer) Science from Purdue University in 1985. His areas of interest include operating systems, distributed computing, networking, and system integration.

ANNE F. KOHUTANYCZ, System Administrator (1985)

Kohutanycz joined the RIACS staff in January, 1985, after holding the positions of Office Manager and Administrative Assistant at CRS Inc., San Jose, California. An active member of the Air Force Reserves since 1978, she is currently serving as the Command Section Administrator for the 349th Field Maintenance Squadron, Travis AFB, California. She received a B.S. in Business Administration from San Jose State University in 1984 and an A.A. in Computer Science from Hartnell College in 1982.

FUNG-FUNG LEE, Student (1987)

Lee is working on his MS degree in the Department of Electrical Engineering at Stanford University. His research is concentrated in parallel computer architecture.

HARRISON LEONG, Research Associate (1986-87)

Leong joined RIACS in October 1986. Previous to this, he did systems programming for Intel's iPSC and postdoctoral work on artificial neural networks at Caltech. He received his PhD in Bioinformation at Caltech in January, 1986 and a BA in Physics from Claremont Men's College in June, 1980. He has one publication in the field of visual psychophysics and a technical report on studying electromagnetic fields of the brain using hypercube concurrent computers. His current interests include developing mathematical models of brain structures and functions and utilizing artificial neural networks to analyze electromagnetic fields of the brain.

BARRY M. LEINER, Senior Staff Scientist (1985)

Leiner came to RIACS in August 1985 from the Defense Advanced Research Projects Agency where he was Assistant Director for C3 Technology in the Information Processing Techniques Office. In that position, he was responsible for a broad range of research programs aimed at developing the technology base for large-scale survivable distributed command, control and communication systems. Prior to that, he was Senior Engineering Specialist with Probe Systems, Assistant Professor of Electrical Engineering at Georgia Tech, and Research Engineer with GTE Sylvania. Dr. Leiner received his BEEE from Rensselaer Polytechnic Institute in 1967 and his M.S. and PhD from Stanford University in 1969 and 1973, respectively. He has done research in a number of areas, including direction finding systems, spread spectrum communications and detection, data compression theory, image compression, and most recently computer networking and its applications. He has published in these areas in both journals and conferences, and received the best paper of the year award in the IEEE Aerospace and Electronic Systems Transactions in 1979 and in the IEEE Communications Magazine in 1984. His current research interests are in distributed systems and how such systems may be used to support scientific research. Dr. Leiner is a Senior Member of the IEEE and a member of ACM, Tau Beta Pi, and Eta Kappa Nu.

EUGENE LEVIN, Scientist & Deputy Director (1983)

Levin came to RIACS in June, 1983. He was Chief Engineer of the Systems and Software Division of System Development Corporation (SDC), where he helped analyze, design, and implement portions of large scale military command, control, and communications systems. Prior to joining SDC, he was Vice President of Culler-Harrison, Inc., where he helped develop an advanced array processor for signal analysis. During 1961-70 he worked at the Aerospace Corporation as Director of the Guidance and Control Subdivision. Dr. Levin has served on the AFIPS Board of Directors and as Chairman of the AIAA Computer Systems Committee. He has over fifty technical publications in various fields of applied mathematics, physics, and computer technology. His education includes an A.B. (1950) and M.A. (1951) in Physics from UCLA, and a PhD (1955) in Mathematics from UCLA. In preparation for his current position, he returned to school and received a BA in Chemistry from the University of Colorado in May 1983. His task assignments are divided between developing advanced architectural concepts for the solution of problems in computational chemistry, and providing technical advice to the NASA Numerical Aerodynamic Simulation (NAS) Project.

ARMANDO E. LOPEZ, Consultant (1987)

Lopez joined RIACS in August, 1986, from NASA Ames Research Center where he had worked as a Research Scientist for thirty five years. His work included nine years in aerodynamic research at the Twelve-Foot Pressure Wind Tunnel, eleven years in Guidance and Navigation and Stability and Control, and fifteen years as a staff assistant to the Flight Systems and Simulation Research Division. He obtained a B.A. in Physics from the University of California, Berkeley in 1951.

WILLIAM LYNCH, Student (1987)

Lynch is working on his MS degree from the Department of Electrical Engineering at Stanford University. His research is in distributed computing systems.

ARI OLLIKAINEN, Scientist (1987)

Ollikainen came to RIACS in August 1987, from General Electric Western Systems of San Jose where as Consulting Systems Engineer he was assigned the technical leadership responsibility as Program Engineer for General Electric's participation in the Numerical Aerodynamic Simulation Program at NASA Ames (1983-1987). He is generally acknowledged for defining and developing the networking strategy and architecture to support remote access by scientific workstations to the supercomputing resources of the Numerical Aerodynamic Simulation Facility at ARC.

Ollikainen has been involved with some facet of large-scale scientific computing since, as a high school student, he took his first programming course at UCLA in the summer of 1960. That FORTRAN class led to employment as a scientific programmer in his first year as an engineering student at UCLA. Eventually it led him to the management of the ARPAnet's Network Measurement Center (1969-1973) and the National Institute of Health's Health Sciences Computing Facility (1974-1977) both located at UCLA.

Having taken a "flyer" in the aircraft chartering and leasing business and bitten by the entrepreneurial urge he moved to Silicon Valley, where, after a stint as manager of computing and communications resources at the Palo Alto lab of Bell Northern Research, he was co-founder and product architect of SYTEK Inc. Leaving Sytek, he provided clients with technical and management consulting in the area of high-performance local area networks as well as serving as a participant on the IEEE Local Area Network Standards Committee.

He has been a catalyst in the development of the NSF funded Bay Area Regional Research Network, BARRNet, and has served as a technical advisor to the National Science Foundation on national networking issues. He is currently active in various technical committees, working groups, and task forces focusing on research and issues related to providing high-performance access to high-performance computing environments from remote user locations.

BRUNO A. OLSHAUSEN, Research Associate (1987)

Olshausen joined RIACS in November, 1987, after receiving his M.S. and B.S. degrees in Electrical Engineering from Stanford University. While at Stanford, he held research assistantships in the Robotics and Computer Vision Lab and in the Perception and Cognition Lab at NASA Ames Research Center, where he worked on computer graphics and image processing programs. As a Hughes Co-op Internship student during the summers 1983-85, Olshausen developed a terrain rendering program and worked on the interpretation of range image data for the Autonomous Land Vehicle project. In 1985, he worked as a Stanford-Krupp Intern at the German Aerospace Research Establishment, where he developed a program for rendering planetary surfaces. Olshausen's current research interest is in neural network approaches to visual pattern recognition.

MICHAEL R. RAUGH, Chief Scientist (1985)

Raugh came to RIACS in January, 1985, from Hewlett-Packard Laboratories of Palo Alto, where he worked in math applications and chaired organizational efforts leading to establishment of the HP Labs Scientific Computing Facility. He received his M.S. and PhD in Mathematics from Stanford University in 1977 and 1979, respectively. He received a B.S. from UCLA in 1962. A former Fulbright/Hayes and National Science Foundation Fellow in mathematics, Raugh has worked as applications and systems programmer at Lawrence Berkeley Laboratory, US Geological Survey, and the Institute for Mathematical Studies in the Social Sciences at Stanford University. He has written data storage and retrieval systems, Monte Carlo simulations, computer assisted instruction programs, program development systems, calibration and digital filtering programs. His central interest is the application of mathematics and computers to modeling physical systems. He is a member of SIAM and Sigma Xi.

DAVID ROGERS, Scientist (1987)

Rogers joined the RIACS staff in August, 1987. He received a B.S. in Chemistry and a B.A. in Applied Mathematics from the University of California - Berkeley, and a PhD from the University of California - Santa Cruz. Following a one year Post Doctoral appointment at the MIT Artificial Intelligence Lab, he spent three years at the University of Michigan working with Dr. Doug Hofstadter. His interests include massive parallelism, punk rock, artificial intelligence, and so-called 'neural-networks'.

MICHAEL SLOCUM, Research Associate (1987)

Slocum joined RIACS in June, 1987 to work on the Telescience Testbed Pilot Program. He has a B.A. in Mathematics (1978) from Kalamazoo College, Michigan and an M.S. in Computer Science (1981) from the University of California, Davis. After completing his masters degree, he spent five years as a Computer Scientist and Engineer at Lawrence Livermore National Laboratory in the Mirror Fusion Test Facility and Advanced Test Accelerator programs, developing supervisory, distributed user interfaces and experimental control systems. Before coming to RIACS, he spent two years as a Systems Programmer and member of the research staff at Stanford University in the Distributed Systems Group doing user interface management system design and implementation. His interests include adaptive, distributed, and user interface management systems, multi-media communication, and object-oriented programming environments.

HELEN STEWART, Project Secretary (1987)

Stewart joined RIACS in September, 1987. She began as a general secretary and currently is project secretary for the Sparse Distributed Memory project. Prior to that she held the position of regional secretary for the McDonnell Douglas Corporation. She is working towards her A.A. in Business at Evergreen College.

RAYMOND TUMINARO, Student (1986)

Tuminaro came to RIACS in June, 1986, while a Ph.D. student at Stanford University. He is working with Tony Chan on the application of multigrid methods to parallel processing systems. During summer 1986, he implemented a multigrid method on the Intel iPSC hypercube, demonstrating the potential performance improvements using that architecture. His central interests include the development of computer algorithms to solve computational fluid dynamics problems on parallel computers. He is presently working on his Ph.D. in the Numerical Analysis program in Computer Science at Stanford. He received a B.S. in Computer Science from Cornell University in 1984. He has been a teaching assistant and research assistant at Stanford and worked as a numerical consultant at the Stanford Linear Accelerator.

AVERY WANG, Student (1987)

Wang is working on his M.S. degree in the Department of Electrical Engineering at Stanford University. His research is in parallel computation.

MONTE ZWEBEN, Visiting Research Associate (1987)

Zweben is currently in the M.S. program of Stanford's Computer Science Department. He received his B.S. in Applied Math/Computer Science and Industrial Management from Carnegie-Mellon University. His current interests are Machine Learning and Planning and Scheduling which are subfields of Artificial Intelligence. He is interested in applying learning techniques to planning systems so that they may improve themselves over time. Before joining RIACS, he was the Group Leader of the MITRE Corporation's AI Research Group where he worked on Natural Language and Scheduling Systems.

APPENDIX A

RIACS PUBLICATION LIST 1987

Books and Book Chapters

1. **Pentti Kanerva,** "Sparse Distributed Memory," (Fall 1988) (Based on 1984 thesis with an additional final chapter dealing with how to incorporate SDM in an autonomous system with sensors and motor controllers.) MIT Press, Cambridge. (In press).

2. **George B. Adams,** "Research Questions for Performance Analysis of Supercomputers," (with G.B. Adams III) in *SUPERCOMPUTING: State of the Art* (A. Lichnewsky & C. Saguez, eds.), North-Holland (1987), pp. 171-184.

3. **Peter Cheeseman,** "A Method of Computing Maximum Entropy Probability Values for Expert Systems," *Maximum-Entropy and Bayesian Spectral Analysis and Estimation Problems,* by the Reidel Publishing Company, pp. 229-240, 1987.

Reviewed Periodicals

4. **Peter J. Denning,** "Security of Data in Networks," *American Scientist,* Vol. 75, No. 1 January-February 1987, pp. 12-14.

5. **B. M. Leiner,** D. L. Nielson, and F. A. Tobagi, "Issues in Packet Radio Network Design," *Proceedings of the IEEE, Special Issue on Packet Radio Networks,* January 1987, pp. 6-20.

6. **Marjory J. Johnson,** "Proof that Timing Requirements of the FDDI Token Ring Protocol are Satisfied," *IEEE Transactions on Communications,* 1987, Vol. COM-35, pp. 620-625.

7. **Peter J. Denning, George B. Adams,** "Research Questions for Performance Analysis of Supercomputers," *Proceedings of the International Seminar on Large-Scale Scientific Computation,* INIRA, February 1987.

8. **Peter J. Denning,** "Evaluating Supercomputers," *American Scientist,* Vol. 75, No. 2, March-April 1987, pp. 130-132.

9. **Kenneth C. Sevcik and Marjory J. Johnson.** "Cycle Time Properties of the FDDI Token Ring Protocol," *IEEE Transactions on Software Engineering,* Vol. SE-13, No. 3, March 1987, pp. 376-385.

10. **Peter J. Denning,** "Multigrids and Hypercubes," *American Scientist,* Vol. 75, No. 3, May-June 1987, pp. 234-238.

11. **Peter J. Denning,** "Computer Modeling of AIDS Epidemiology," *American Scientist,* Vol 75., No. 4, July-August 1987, pp. 347-352.

12. **Peter J. Denning,** "Baffling Big Brother," *American Scientist,* Vol. 75, No. 5. September-October 1987, pp. 464-466.

13. **Peter J. Denning,** "A New Paradigm for Science," *American Scientist,* Vol. 75, No. 6, November-December 1987, pp. 572-573.

14. **George B. Adams III,** D. P. Agrawal, and H. J. Siegel, "A Survey and Comparison of Fault-Tolerant Multistage Interconnection Networks," *IEEE Computer,* to appear.

15. J. P. Crutchfield and **Julian E. Gomez,** "Chaos, Dynamics, and Natural Phenomema," *ACM Transactions on Graphics,* to appear.

Reviewed Conference Proceedings

16. **Tony Chan,** Ray S. Tuminaro, "Design and Implementation of Parallel Multigrid Algorithms," *Proceedings of the Third Copper Mountain Conference on Multigrid Methods,* April 1987.

17. **James D. Keeler,** "Comparison of Information Capacity of Hebbian Neural Network Models," *Proceeding of the IEEE Conference on Neural Networks,* June 1987.

18. **E. Levin,** C.K. Eaton, and B. Young, "Scaling of Data Communications for an Advanced Supercomputer Network," *IFIP/IEEE/ITC Third International Conference of Data Communications and Their Performance,* June 1987.

19. **Peter Cheeseman,** Matthew Self, "Bayesian Prediction for Artificial Intelligence," *Uncertainty in Artificial Intelligence,* sponsored by American Association of Artificial Intelligence (AAAI), Martin Marietta (Baltimore Laboratories), and Advanced Decision Systems, pp. 61-69, July 1987.

20. **Julian Gomez,** "Further Comments on Event Driven Animation" part of tutorial "Computer Animation: 3-D Motion Specification and Control," *Annual ACM/SIGGRAPH Conference,* July 1987.

21. **Barry M. Leiner,** "Access Control and Privacy in Large Distributed Systems," *IEEE Computer Communications Workshop,* September 1987.

22. **Peter J. Denning,** "Software for Parallel Architectures," *Proceedings for the AIAA Computers in Aerospace VI Conference,* October 1987.

23. **Peter Cheeseman,** Jim Kelly, Matthew Self, and John Stutz, "Automatic Bayesian Induction of Classes," *Conference Proceedings of the Second Annual Artificial Intelligence Research Forum,* NASA Ames Research Center, pp. 224-239, November 1987.

24. **Tony Chan, Ray S. Tuminaro,** "A Survey of Parallel Multigrid Methods," *Proceedings of the American Society of Mechanical Engineers Winter Annual Meeting,* December 1987.

25. **E. Levin,** F. Preston, "The Role of Specialized Processors in the NAS Program: Retrospective/Prospective," *Proceedings of the 1988 Multiprocessor and Array Conference,* February 1988, pp. 47-51, (This was prepared and accepted in 1987).

Other Conference Proceedings

26. **Marjory J. Johnson,** "FDDI Analysis at NASA Ames," *ANSI FDDI Standards Meeting,* January 1987.

27. **Barry M. Leiner,** "Packet Radio Networks," *Proceedings of INFOCOM87,* April 1987.

28. **Matt Bishop,** "Sharing Accounts," *Proceedings of the Large Installation System Administrator's Workshop,* April 1987 (to appear).

29. **James D. Keeler,** "Capacity for Patterns and Sequences in Kanerva's SDM as Compared to Other Associative Memory Models," *Proceedings of the Neural Information Processing Systems Conference,* November 1987.

Other Articles and Newsletter Items

30. **Peter J. Denning,** "The Lost Art of Self Defense," Editorial, *Communications of the ACM,* February 1987, pp. 110-111.

31. **Matt Bishop,** "Array Names and Pointers," *The C Journal,* Vol. 3, No. 1, Spring 1987.

32. **Matt Bishop,** "How to Write a Setuid Program," *;login;* January-February 1987, pp.6-20

33. **Matt Bishop,** "File Protection in UNIX," *The DEC Professional, Special Edition,* June 1987, pp. 44-48.

34. **Barry M. Leiner,** "Telescience, Space Station, and the Telescience Testbed," *README; Sun Microsystems Users Group,* Summer 1987.

35. **Peter J. Denning,** "Paradigms Crossed," Editorial, *Communications of the ACM,* October 1987, pp. 808-809.

36. **Matt Bishop,** "Storage in C," *The C Journal,* to appear.

37. **Matt Bishop,** "How to Use the USENET Effectively," submitted for inclusion in the News B2.10.3 release, and submitted to *USENIX.*

38. **Peter J. Denning,** "Selling Science," Editorial, *Communications of the ACM,* to appear February 1988.

39. **Peter J. Denning,** "Policy Discussion on Research Papers," Editorial, *Communications of the ACM,* to appear February 1988.

Technical Reports

1. **Barry M. Leiner,** *Networking Requirements for Scientific Research,* TR 87.1, Jan. 1987, 24 pp.

2. **Peter J. Denning,** *Evaluating Supercomputers,* TR 87.2, January 1987, 16 pp.

3. **Peter J. Denning, Robert L. Brown, Barry M. Leiner,** *The RIACS Scientist's Workbench,* TR 87.3, February 1987, 13 pp.

4. **Tony F. Chan,** Diana C. Resasco, *Hypercube Implementation of Domain Decomposed Fast Poisson Solvers,* TR 87.5, November 1986, 9 pp.- 15 pp.

5. **Matt Bishop,** *The RIACS Mail System,* TR 87.6, February 1987, 24 pp.

6. **Barry M. Leiner,** *OSSA Networking Architecture,* TR 87.7, March 1987, 26 pp.

7. **Peter J. Denning,** *Multigrids and Hypercubes,* TR 87.8, March 1987, 12 pp.

8. **George B. Adams,** *Multistage Interconnection Networks,* TR 87.9, 64 pp.

9. **Matt Bishop,** *A Mechanism for Sharing Accounts,* TR 87.10, March 1987, 29 pp.

10. **Robert L. Brown, Peter J. Denning,** *Evaluation of a Parallel Program Architecture,* TR 87.11, May 1987, 18 pp.

11. **Barry M. Leiner,** *Telescience Testbed Pilot Program,* TR 87.12, May 1987, 42 pp.

12. **Eugene Levin,** *Transport Collision Integrals for O-O, N-N, O+_O and N+_N,* TR 87.13, May 1987.

13. **Peter J. Denning,** *Computer Models of AIDS Epidemiology,* TR 87.14, May 1987, 15 pp.

14. **Marjory J. Johnson,** *Network Protocols for Real Time Applications,* TR 87.15, May 1987, 15 pp.

15. **Julian E. Gomez,** *Comments on Event Driven Computer Animation,* TR 87.16, May 1987, 33 pp.

16. **Robert L. Brown,** *Two Demonstrators and a Simulator for a Sparse, Distributed Memory,* TR 87.17, June 1987, 14 pp.

17. **Matt Bishop,** *A Fast Version of the DES and Password Encryption Algorithm,* TR 87.18, July 1987, 24 pp.

18. **Peter J. Denning,** *Baffling Big Brother,* TR 87.19, July 1987, 12 pp.

19. **Peter J. Denning,** *Software for Parallel Architectures,* TR 87.20, July 1987, 6 pp.

20. **Tony F. Chan,** Ray S. Tuminaro, *Design and Implementation of Parallel Multigrid Algorithms,* TR 87.21, August 1987, 18 pp.

21. **Tony F. Chan,** Ray S. Tuminaro, *A Survey of Parallel Multigrid Algorithms,* TR 87.22, August 1987, 20 pp.

22. **Peter J. Denning,** *Paradigms Crossed,* TR 87.23, August 1987, 8 pp.

23. **Peter J. Denning,** *A New Paradigm for Science,* TR 87.24, September 1987, 10 pp.

24. **Maria L. Gallagher,** *Telescience Testbed Kickoff Meeting Minutes,* TR 87.25, September 1987, 26 pp.

25. **Maria L. Gallagher,** *Telescience Testbed Pilot Program Quarterly Report,* TR 87.26, October 1987, 35 pp.

26. **Fung F. Lee and George B. Adams,** *Implementation of Lattice Gas Cellular Automata on Hypercubes,* TR 87.27, October 1987, 21 pp.

27. **Peter J. Denning,** *Deadlocks,* TR 87.28, October 1987, 9 pp.

28. **James D. Keeler,** *Capacity for Patterns and Sequences in Kanerva's SDM as Compared to Other Associative Memory Models,* TR 87.29, November 1987, 10 pp.

29. **Erol Gelenbe,** *Performance Analysis Approximations for Parallel Processing of Concurrent Tasks,* TR 87.30, December 1987, 21 pp.

30. **Barry M. Leiner, Maria L. Gallagher,** *Telescience Testbed Pilot Program Second Quarterly Report,* TR 87.31, December 1987, 54 pp.

APPENDIX B

Talks and Presentations by RIACS Staff in 1987

1. Marjory Johnson, "FDDI Analysis at NASA Ames," FDDI ANSI standards meeting, January 1987.
2. Peter J. Denning, "Status of Task Force on Core of Computer Science," ACM Computer Science Conference, February 17, 1987.
3. Pentti Kanerva, "Sparse, Distributed Memory for Patterns and Sequences," Institute for Nonlinear Science Seminar on Neural Networks, March 1987.
4. Peter J. Denning, "Evaluating Supercomputers," U California Berkeley, April 16, 1987.
5. Julian Gomez, "Models of Computer Animation Systems," Technische Hochschule, Darmstadt, West Germany, April 1987.
6. Peter J. Denning, "What is Computer Science?" U California Davis, May 21, 1987.
7. Pentti Kanerva, "Sparse, Distributed Memory," Psychology/Computer Science Seminar on Computational Models of Perception and Cognition, May 1987.
8. E. Levin, Harry Partridge, and J.R. Stallcop, "High Temperature Transport Properties of Air," AIAA Paper 87-1632, Presented at the AIAA 22nd Thermo-Physics Conference, June 8-10, 1987.
9. Pentti Kanerva, "A Long-Term Memory for an Autonomous Adaptive System," Ninth Annual Cognitive Science Society Conference Symposium, July 1987.
10. Barry M. Leiner, "Telescience and Advanced Technologies," ACM SIGCOMM87, August 1987.
11. Julian Gomez, "Workstations in the Next Decade," Poster at NASA Spatial Displays and Spatial Instruments Symposium/Workshop, Asilomar, CA, September 1987.
12. Peter J. Denning, "Software for Parallel Architectures," AIAA Conference, October 8, 1987.
13. Peter J. Denning, "History of ACM: 1975-82," Fall Joint Computer Conference, October 27, 1987.
14. Julian Gomez, "Computer Animation," DeAnza College Computer Animation Class, October 1987.
15. Barry M. Leiner, Jim Weiss, "Telescience Testbedding: An Implementation Approach," *1st Meeting of International Taskforce on the Scientific Uses of Space Station,* October 1987.
16. Barry M. Leiner, "Scientific Networking: Challenges and Opportunities," Dinner Speaker, Data Systems User Working Group, November 1987.
17. Vicki Johnson, "Telescience Testbedding," UCLA Space Physics Dept., November 1987.
18. Tony Chan, Ray S. Tuminaro, Dennis Jasperson, Eric Barczcz, "Solution of the Euler Equations on a Hypercube," Poster at the SIAM Meeting on Parallel Processing, December 1987.

RIACS
Mail Stop 230-5
NASA Ames Research Center
Moffett Field, CA 94035
(415) 694-6363

The Research Institute for Advanced Computer Science
is operated by
Universities Space Research Association
The American City Building
Suite 212
Columbia, MD 21044
(301) 730-2656